PROGRESO O AMENAZA

MELANIE MITCHELL

INTELIGENCIA ARTIFICIAL

Guía para **seres pensantes**



INTELIGENCIA ARTIFICIAL

Guía para **seres pensantes**

MELANIE MITCHELL

Traducción de

María Luisa Rodríguez Tapia

Capitán Swing

Prólogo

Aterrorizados

Da la impresión de que los ordenadores están volviéndose cada vez más inteligentes a una velocidad alarmante, pero algo que todavía no han aprendido a hacer es apreciar la ironía. En eso pensaba yo hace unos años cuando, de camino a un debate sobre inteligencia artificial (IA), me perdí en la capital de las búsquedas: Googleplex, la sede mundial de Google en Mountain View, California. No solo eso; me perdí dentro del edificio de Google Maps. Ironía al cuadrado.

El edificio de Maps en sí no me había costado encontrarlo. Había un coche de Google Street View aparcado junto a la puerta principal, un enorme apéndice coronado por una cámara en forma de balón de fútbol rojo y negro colocada sobre el techo. Pero cuando entré, con la llamativa tarjeta de «visitante» que me habían dado los de seguridad, deambulé sin rumbo, abochornada, por los laberintos de cubículos llenos de empleados de Google que, con auriculares en las orejas, tecleaban en sus ordenadores Apple. Tras una búsqueda azarosa (y sin mapa), por fin encontré la sala de conferencias asignada para la reunión, que debía durar todo el día, y me uní a la gente allí congregada.

El encuentro, que se celebró en mayo de 2014, había sido organizado por Blaise Agüera y Arcas, un joven informático que acababa de dejar un alto cargo en Microsoft para incorporarse a Google con el fin de ayudar a dirigir

las investigaciones sobre inteligencia artificial. Google nació en 1998 con un «producto»: un sitio web que empleaba un método nuevo y extraordinariamente logrado para buscar en internet. Con los años, Google ha evolucionado hasta convertirse en la empresa tecnológica más importante del mundo, y hoy ofrece una gran variedad de productos y servicios, como Gmail, Google Docs, Google Translate, YouTube, Android y muchos más que es probable que usted utilice a diario, así como otros de los que seguramente poca gente ha oído hablar.

Los fundadores de Google, Larry Page y Sergey Brin, llevan mucho tiempo dando vueltas a la idea de crear una inteligencia artificial en los ordenadores, hasta el punto de que se ha convertido en uno de los principales objetivos de la compañía. En la última década, Google ha contratado a numerosos expertos en IA, entre ellos Ray Kurzweil, un conocido inventor y polémico futurista que promueve una idea a la que se ha llamado «Singularidad de la IA»: el momento, en un futuro próximo, en el que los ordenadores serán más inteligentes que los humanos. Google contrató a Kurzweil para que pudiera hacer realidad esta idea. En 2011, la empresa creó un grupo de investigación sobre IA llamado Google Brain; posteriormente ha adquirido una enorme variedad de empresas emergentes de IA, todas con nombres optimistas: Applied Semantics, DeepMind y Vision Factory, entre otras.

En resumen, Google ha dejado de ser solo un portal de búsquedas por internet. Se está convirtiendo rápidamente en una empresa de IA aplicada. La IA es el pegamento que une los diversos productos, servicios y trabajos de investigación de Google y su empresa matriz, Alphabet. La aspiración suprema de la empresa queda reflejada en la declaración de objetivos original de su grupo de DeepMind: «Resolver la inteligencia y utilizarla para resolver todo lo demás».[1]

AI y *GEB*

Tenía muchas ganas de asistir a una reunión sobre IA en Google. Llevaba trabajando en varios aspectos de la IA desde que hice el posgrado, en los años ochenta, y me había impresionado mucho todo lo que Google había conseguido. También pensaba que podía aportar alguna idea. Pero tengo que reconocer que acudí solo como acompañante. El objetivo de la reunión era que un grupo escogido de investigadores de IA de Google pudiera mantener una conversación con Douglas Hofstadter, una leyenda del mundo de la IA y autor de un famoso libro con el críptico título de *Gödel, Escher, Bach. Un eterno y grácil bucle* o, para ser más breves, *GEB*.[2] Cualquier informático o aficionado a la informática probablemente ha oído hablar de él, lo ha leído o ha intentado leerlo.

Escrito en los años setenta, *GEB* da salida a las numerosas pasiones intelectuales de Hofstadter: matemáticas, arte, música, lenguaje, humor y juegos de palabras, reunidas para abordar las trascendentales preguntas sobre cómo es posible que la inteligencia, el conocimiento y la propia conciencia, que son experiencias tan fundamentales del ser humano, surjan del sustrato no inteligente y no consciente que son las células biológicas. También aborda de qué manera los ordenadores podrían acabar adquiriendo inteligencia y autoconciencia. Es un libro extraordinario; no conozco ningún otro que se parezca lo más mínimo. Pese a no ser fácil de leer, fue un éxito de ventas y ganó el Premio Pulitzer y el National Book Award. Desde luego, *GEB* fue un libro que impulsó a muchos jóvenes a dedicarse a la IA. Entre ellos, yo misma.

A principios de los ochenta me había licenciado en Matemáticas y vivía en Nueva York; daba clases en una escuela preparatoria, me sentía infeliz y trataba de averiguar qué quería hacer verdaderamente con mi vida. Descubrí *GEB* gracias a una crítica muy elogiosa en *Scientific American* y me apresuré a comprar el libro. Lo devoré en las semanas siguientes y me convencí no solo de que quería dedicarme a la investigación sobre IA, sino

de que quería trabajar con Douglas Hofstadter. Nunca un libro me había causado tanta impresión ni había tenido tan claro mi rumbo profesional.

En aquella época, Hofstadter era catedrático de Informática en la Universidad de Indiana, y mi utópico plan era solicitar el ingreso en el programa de doctorado en Informática, presentarme y convencer a Hofstadter para que me aceptara como estudiante. Solo había un pequeño inconveniente: nunca había asistido a ninguna clase de Informática. Había crecido entre ordenadores; mi padre era ingeniero informático en una empresa tecnológica de los años sesenta y, por puro *hobby*, construyó un servidor en el cuarto de estar familiar. El ordenador Sigma 2, del tamaño de un frigorífico, tenía un botón magnético que proclamaba «Rezo en FORTRAN»; y yo, de niña, estaba casi convencida de que de verdad rezaba en silencio por las noches, mientras la familia dormía. Como crecí en los años sesenta y setenta, aprendí un poco del lenguaje de moda en cada periodo: FORTRAN, luego BASIC, después Pascal; pero no sabía prácticamente nada de técnicas de programación propiamente dichas, y mucho menos de todas las demás cosas que debe saber alguien que quiere hacer un posgrado en Informática.

Para acelerar mi plan, al acabar el curso académico dejé mi puesto de profesora, me mudé a Boston y me matriculé en cursos de iniciación a la informática para prepararme para mi nueva carrera. Un día, a los pocos meses de empezar mi nueva vida, mientras estaba en el campus del Instituto Tecnológico de Massachusetts (MIT) esperando a que empezara una clase, vi un cartel que anunciaba una conferencia que iba a pronunciar Douglas Hofstadter allí mismo dos días después. Me pareció increíble la suerte que había tenido. Fui a la conferencia, esperé mucho tiempo mi turno en medio de una multitud de admiradores y conseguí hablar con él. Me enteré de que estaba en pleno año sabático en el MIT y que después iba a marcharse de Indiana para trasladarse a la Universidad de Míchigan, en Ann Arbor.

Para no extenderme: después de perseguirlo con cierta insistencia, convencí a Hofstadter para que me aceptara como ayudante de investigación, primero durante un verano y luego, durante los seis años siguientes, como alumna de posgrado, hasta obtener el doctorado en Informática por la Universidad de Míchigan. Hofstadter y yo hemos mantenido un estrecho contacto todos estos años y hemos debatido mucho sobre IA. Él conocía mi interés por las investigaciones de Google sobre IA y tuvo la amabilidad de invitarme a acompañarle a la reunión en la sede de la empresa.

El ajedrez y el primer germen de duda

El grupo de la semiescondida sala de conferencias estaba formado por una veintena de ingenieros de Google (además de Douglas Hofstadter y yo), todos ellos miembros de diversos equipos de IA de Google. La reunión comenzó, como era habitual, con la presentación de cada uno. Varios señalaron que lo que había impulsado su carrera en el campo de la IA había sido la lectura de *GEB* cuando eran jóvenes. Todos estaban entusiasmados y con curiosidad por saber lo que iba a decir el legendario Hofstadter sobre la IA. Él se levantó y tomó la palabra. «Tengo unas cuantas observaciones sobre la investigación de la IA en general y sobre lo que se hace aquí en Google en particular —su voz se llenó de pasión—. Estoy aterrorizado. Aterrorizado».

Y continuó.[3] Comento que, cuando en los años setenta había empezado a trabajar en la IA, aquella era una perspectiva que, si bien resultaba apasionante, parecía estar tan lejos de hacerse realidad que no había «ningún peligro en el horizonte, ninguna sensación de que fuera verdaderamente a ocurrir». Crear máquinas con una inteligencia similar a la humana era una aventura intelectual de calado, un proyecto de investigación a largo plazo para el que se decía que faltaban por lo menos «cien premios nobel».[4] Hofstadter creía que, en principio, la IA era posible: «El

“enemigo” eran personas como John Searle, Hubert Dreyfus y otros escépticos, que decían que era imposible. No entendían que un cerebro es un trozo de materia que se rige por leyes físicas y que el ordenador puede simular cualquier cosa: el nivel de neuronas, los neurotransmisores, etc. En teoría, se puede hacer». De hecho, las ideas de Hofstadter sobre la simulación de la inteligencia en varios planos —desde las neuronas hasta la conciencia— se abordaban con gran detalle en *GEB* y eran la base de sus investigaciones desde hacía décadas. Sin embargo, en la práctica, Hofstadter había pensado hasta poco tiempo antes que no había ninguna posibilidad de que una IA general «a nivel humano» llegara a hacerse realidad en vida suya (ni siquiera en la de sus hijos), así que no le preocupaba mucho.

Casi al final de *GEB*, Hofstadter presentaba una lista de «Diez preguntas y conjeturas» sobre la inteligencia artificial. He aquí una de ellas: «¿Habrá programas de ajedrez capaces de vencer a todo el mundo?» La hipótesis de Hofstadter era que no. «Quizá haya programas capaces de ganar a cualquiera en una partida, pero no serán exclusivamente programas de ajedrez. Serán programas de inteligencia general».[5]

En la reunión de 2014 en Google, Hofstadter reconoció que se había «equivocado por completo». El rápido perfeccionamiento de los programas de ajedrez en los años ochenta y noventa había sembrado el primer germen de duda en su valoración de las posibilidades de la IA a corto plazo. Aunque el pionero de la IA Herbert Simon había predicho en 1957 que habría un programa de ajedrez campeón del mundo «en un plazo de diez años», a mediados de los años setenta, cuando Hofstadter estaba escribiendo *GEB*, los mejores programas informáticos de ajedrez no alcanzaban más que el nivel de un buen (no un gran) aficionado. Hofstadter se había hecho amigo de Eliot Hearst, campeón de ajedrez y profesor de Psicología, que había escrito mucho sobre las diferencias entre un jugador humano experto y un programa informático de ajedrez. Los experimentos

demostraban que un jugador humano, para decidir una jugada, utiliza el reconocimiento rápido de patrones en el tablero, en vez de la búsqueda exhaustiva y a las bravas que emplean todos los programas de ajedrez. Durante una partida, los mejores jugadores humanos perciben una disposición de las piezas como «un tipo de posición» concreto que exige «un tipo de estrategia» determinado. Es decir, esos jugadores saben identificar rápidamente configuraciones y estrategias concretas como casos concretos de conceptos teóricos. Hearst afirmaba que, sin esa capacidad general de percibir patrones y reconocer conceptos abstractos, los programas de ajedrez nunca alcanzarían el nivel de los mejores jugadores humanos. A Hofstadter le convencieron los argumentos de Hearst.

Sin embargo, en los años ochenta y noventa, los programas de ajedrez mejoraron de golpe, sobre todo por la gran velocidad que adquirieron los ordenadores. Aun así, los mejores programas seguían jugando de una forma muy poco humana: realizando una amplia búsqueda antes de decidir la siguiente jugada. A mediados de los noventa, la máquina Deep Blue de IBM, con un *hardware* especializado para jugar al ajedrez, alcanzó el nivel de Gran Maestro; en 1997 derrotó al entonces campeón mundial de ajedrez, Garry Kasparov, en un desafío a seis partidas. La maestría en el ajedrez, antes considerada la cumbre de la inteligencia humana, había sucumbido a la fuerza bruta.

La música, el bastión de la humanidad

Aunque la victoria de Deep Blue generó muchas muestras de inquietud en la prensa por el auge de las máquinas inteligentes, la «auténtica» IA parecía todavía muy lejana. Deep Blue sabía jugar al ajedrez, pero no podía hacer nada más. Hofstadter se había equivocado sobre eso, pero seguía considerando válidas las demás conjeturas de *GEB*, sobre todo la primera que había mencionado:

PREGUNTA: ¿Alguna vez un ordenador compondrá música llena de belleza?

CONJETURA: Sí, pero falta mucho.

Y Hofstadter proseguía:

> La música es un lenguaje de emociones y, hasta que los programas no tengan emociones tan complejas como las nuestras, es imposible que un programa componga algo bello. Puede haber «falsificaciones» —imitaciones superficiales de la sintaxis de otra música anterior—, pero, pese a lo que en principio podría parecer, en la expresión musical hay mucho más que lo que pueden captar las reglas sintácticas. […] Pensar […] que quizá pronto seamos capaces de ordenar a una «caja de música» de mesa, preprogramada, fabricada en serie y comprada por correo por veinte dólares, que saque de sus circuitos estériles unas obras que podrían haber compuesto Chopin o Bach si hubieran vivido más tiempo es valorar de manera grotesca, vergonzosa y errónea las profundidades del espíritu humano.[6]

Hofstadter decía que esta conjetura era «una de las partes más importantes de *GEB*; me habría apostado la vida por ella».

A mediados de los noventa, la seguridad de Hofstadter en su opinión sobre la IA volvió a tambalearse, esta vez muy en serio, cuando se encontró con un programa escrito por un músico, David Cope. El programa se llamaba Experimentos en Inteligencia Musical (en inglés, EMI, pronunciado tal cual). Cope, compositor y profesor de Música, había creado originalmente EMI para facilitar su propio trabajo mediante la composición automática de piezas según su estilo personal. Sin embargo, EMI se hizo famoso por crear obras al estilo de compositores clásicos como Bach y Chopin. EMI compone siguiendo una gran cantidad de reglas establecidas por Cope con el fin de construir una sintaxis general de composición. Estas reglas se aplican a numerosos ejemplos de la obra de un compositor concreto para crear una pieza nueva «al estilo» de ese compositor.

Volviendo a nuestra reunión en Google, Hofstadter habló con una emoción extraordinaria de los contactos que había tenido con EMI:

> Me senté al piano y toqué una de las mazurcas de EMI «al estilo de Chopin». No sonaba exactamente como una obra suya, pero se parecía tanto a algo de Chopin y a una obra musical coherente que me perturbó en lo más hondo. Desde niño, la música me emociona y me conmueve hasta la médula. Y escucho cada pieza amada como un mensaje directo desde el corazón emocional del ser humano que la compuso. Es como si me permitiera entrar en lo más

> íntimo de su alma. Y tengo la sensación de que no hay nada más humano en el mundo que esa expresión musical. Nada.

Después, Hofstadter habló de una conferencia que había pronunciado en la prestigiosa Escuela de Música Eastman en Rochester, Nueva York. Tras describir EMI, Hofstadter pidió a un pianista que tocara dos piezas y propuso al público —entre el que había varios profesores de Teoría Musical y Composición— que adivinara cuál de las dos era una mazurca (poco conocida) de Chopin y cuál había sido compuesta por EMI. Como explicó más tarde un miembro del público: «La primera mazurca tenía gracia y encanto, pero no la enorme inventiva y fluidez de un "verdadero Chopin". […] Estaba claro que la segunda era el Chopin auténtico, con una melodía lírica, grandes y elegantes modulaciones cromáticas y una forma natural y equilibrada».[7] Es lo que pensaron muchos de los profesores presentes, que, para asombro de Hofstadter, votaron que la primera pieza era de EMI y la segunda del «verdadero Chopin». Las respuestas acertadas eran las contrarias.

En la sala de conferencias de Google, Hofstadter hizo una pausa y nos miró a la cara. Nadie dijo una palabra. Por fin volvió a hablar. «EMI me aterrorizó. Me aterrorizó. Lo detesté, me pareció una horrible amenaza. Una amenaza que podía destruir lo que más valoraba de la humanidad. Creo que EMI fue el ejemplo supremo de los temores que me provoca la inteligencia artificial».

Google y la Singularidad

Luego, Hofstadter habló de la enorme ambivalencia que sentía sobre lo que Google estaba tratando de hacer con la IA: coches sin conductor, reconocimiento del habla, comprensión del lenguaje natural, traducción entre idiomas, arte generado por ordenador, composición musical y muchas más cosas. La preocupación de Hofstadter se agudizaba porque Google había adoptado las ideas de Ray Kurzweil sobre la Singularidad, según las

cuales la IA, impulsada por su capacidad de perfeccionarse y educarse a sí misma, pronto alcanzaría a la inteligencia humana y, después, la superaría. Parecía que Google estaba haciendo todo lo posible para acelerar esa visión. Aunque Hofstadter tenía serias dudas sobre la premisa de la Singularidad, reconoció que las predicciones de Kurzweil le inquietaban. «Me aterrorizaban las posibilidades. Era muy escéptico, pero, al mismo tiempo, pensaba que, si bien su escala temporal podía estar equivocada, quizá tuviera razón. Y entonces nos pillaría totalmente desprevenidos. Pensaremos que no pasa nada y, de repente, antes de que nos demos cuenta, los ordenadores serán más inteligentes que nosotros».

Si sucede esto, «nos sustituirán. Seremos reliquias. Nos dejarán tirados».

Y añadió: «Quizá suceda, pero no quiero que suceda pronto. No quiero que mis hijos se queden tirados».

Hofstadter terminó su charla con una referencia directa a los ingenieros de Google que estaban presentes y le escuchaban atentamente: «Me parece muy aterrador, muy preocupante, muy triste, y me parece terrible, horroroso, extraño, desconcertante, que la gente se precipite a ciegas y delirantemente a crear estas cosas».

¿Por qué está aterrorizado Hofstadter?

Miré a mi alrededor. La gente parecía confusa, incluso avergonzada. Para aquellos investigadores de IA de Google, todo aquello no les parecía aterrador en absoluto. De hecho, no era nada nuevo. Cuando Deep Blue derrotó a Kasparov, cuando EMI empezó a componer mazurcas al estilo de Chopin y cuando Kurzweil escribió su primer libro sobre la Singularidad, muchos de aquellos ingenieros estaban en el instituto, probablemente leyendo *GEB* y encantados con él, aunque sus predicciones sobre IA se hubieran quedado un poco anticuadas. Si trabajaban en Google era precisamente para hacer realidad la IA, no a cien años vista, sino ya, lo antes posible. No entendían por qué Hofstadter estaba tan nervioso.

Los que trabajan en el campo de la IA están acostumbrados a los temores de las personas ajenas a él, seguramente influidas por todas las películas de ciencia ficción en las que aparecen máquinas superinteligentes que se vuelven malvadas. Los investigadores también conocen bien el miedo a que una IA cada vez más sofisticada sustituya a los seres humanos en ciertos trabajos, a que la IA aplicada a grandes conjuntos de datos pueda anular la privacidad y permitir una sutil discriminación y a que los sistemas de IA mal comprendidos y autorizados a tomar decisiones autónomas puedan causar estragos.

Pero el terror de Hofstadter se debía a algo totalmente distinto. No era miedo a que la IA acabara siendo demasiado inteligente, demasiado invasiva, demasiado maligna o incluso demasiado útil. Lo que le aterraba era que la inteligencia, la creatividad, las emociones e incluso la propia conciencia fueran demasiado fáciles de crear, que los aspectos de la humanidad que más valiosos le parecían acabaran siendo una mera «serie de trucos», que un conjunto superficial de algoritmos de fuerza bruta pudiera explicar el espíritu humano.

Como dejaba muy claro en *GEB*, Hofstadter tiene la firme convicción de que la mente, con todas sus características, surge en su totalidad del sustrato físico del cerebro y el resto del cuerpo, además de a través de la interacción del cuerpo con el mundo físico. No hay nada inmaterial o incorpóreo que esté oculto. Su preocupación, en realidad, es la complejidad. Teme que la IA nos enseñe que las cualidades humanas que más valoramos son lamentablemente fáciles de mecanizar. Como me explicó Hofstadter después de la reunión, refiriéndose a Chopin, Bach y otros modelos de humanidad: «Si unas mentes como las suyas, de infinita sutileza y complejidad y hondura emocional, se pudieran trivializar mediante un pequeño chip, se destruiría mi idea de lo que es la humanidad».

Me siento confusa

Tras las palabras de Hofstadter, hubo un breve debate en el que los asistentes, desconcertados, le pidieron que explicara mejor sus temores sobre la IA y sobre Google en particular. Pero seguían existiendo problemas de comunicación. La reunión continuó con presentaciones de proyectos, debates en grupo, pausas para el café, lo de siempre; y ninguna de las cosas que se dijeron tenía realmente nada que ver con las observaciones de Hofstadter. Casi al final del encuentro, Hofstadter preguntó a los participantes qué pensaban sobre el futuro inmediato de la IA. Varios investigadores de Google predijeron que en los próximos treinta años seguramente aparecería una IA general tan capaz como un humano, en gran parte gracias a los avances de Google en el método del «aprendizaje profundo» inspirado en el cerebro.

Salí de la reunión muy confundida. Sabía que a Hofstadter le preocupaban algunas de las cosas que había escrito Kurzweil sobre la Singularidad, pero hasta entonces nunca había visto hasta qué punto estaba conmovido y angustiado. También sabía que Google estaba impulsando las investigaciones sobre IA, pero me sorprendió lo optimistas que eran varios asistentes sobre el tiempo que iba a tardar la IA en alcanzar una capacidad «humana» general. Hasta entonces, mi opinión era que la IA había progresado mucho en algunos ámbitos muy concretos, pero que estaba todavía muy lejos de alcanzar la amplia inteligencia general de los humanos; no lo iba a conseguir en un siglo, y mucho menos en treinta años. Y pensaba que quienes creían lo contrario estaban infravalorando enormemente la complejidad de la inteligencia humana. Había leído los libros de Kurzweil y me habían parecido bastante ridículos. Sin embargo, todos los comentarios que había oído en la reunión, de personas a las que respetaba y admiraba, me obligaron a examinar con una mirada crítica mis opiniones. Me parecía que los investigadores de la IA estaban minusvalorando a los humanos, pero ¿no estaría yo subestimando también el poder y las posibilidades de la IA actual?

En los meses posteriores, empecé a prestar más atención al debate sobre estas cuestiones. Empecé a darme cuenta de la montaña de artículos, entradas de blog y libros enteros de autores destacados que de repente nos estaban diciendo que debíamos empezar a preocuparnos ya por los peligros que representaba la IA «sobrehumana». En 2014, el físico Stephen Hawking proclamó: «El desarrollo de la inteligencia totalmente artificial podría significar el fin de la raza humana».[8] Ese mismo año, el empresario Elon Musk, fundador de las empresas Tesla y SpaceX, afirmó que la inteligencia artificial era probablemente «la mayor amenaza contra nuestra existencia» y que «con la inteligencia artificial estamos invocando al demonio».[9] El cofundador de Microsoft, Bill Gates, se mostró de acuerdo: «Coincido con lo que dicen Elon Musk y otros a este respecto y no entiendo por qué hay gente que no está preocupada».[10] Sorprendentemente, el libro del filósofo Nick Bostrom *Superinteligencia*, sobre los posibles riesgos de que las máquinas acaben siendo más inteligentes que los humanos, fue todo un éxito de ventas, a pesar de su estilo árido y pesado.

Otros destacados pensadores discrepaban. Sí, decían, debemos garantizar que los programas de IA sean seguros y no puedan hacer daño a los seres humanos, pero las afirmaciones de que puede haber una IA sobrehumana a corto plazo son muy exageradas. El empresario y activista Mitchell Kapor opinó: «La inteligencia humana es un fenómeno maravilloso, sutil y mal conocido. No hay peligro de que se duplique a corto plazo».[11] El experto en robótica y exdirector del laboratorio de IA del MIT Rodney Brooks coincidió cuando dijo que «sobrevaloramos enormemente las capacidades de las máquinas, las actuales y las de las próximas décadas».[12] El psicólogo e investigador sobre IA Gary Marcus llegó a afirmar que, en el intento de crear una «IA fuerte» —es decir, una IA general tan capaz como un humano—, «no ha habido casi ningún avance».[13]

Podría seguir mucho tiempo citando declaraciones contrapuestas. En resumen, lo que he descubierto es que el campo de la IA está lleno de

confusión. O se ha avanzado mucho, o casi nada. O estamos a un tiro de piedra de la «verdadera» IA, o todavía faltan siglos. La IA resolverá todos nuestros problemas, nos dejará a todos sin trabajo, destruirá la raza humana o degradará nuestra humanidad. Es una campaña noble o una forma de «invocar al demonio».

De qué trata este libro

Este libro nació de mi intento de comprender la verdadera situación de la inteligencia artificial: qué pueden hacer hoy los ordenadores y qué podemos esperar de ellos en las próximas décadas. Los estimulantes comentarios de Hofstadter en la reunión de Google fueron una especie de llamada de atención para mí, igual que la seguridad con la que respondieron los investigadores de Google sobre el futuro inmediato de la IA. En los capítulos que siguen, voy a tratar de explicar en qué punto se encuentra la IA y de aclarar sus diferentes —y a veces contradictorios— objetivos. Para ello, examinaré cómo actúan realmente algunos de los sistemas de IA más importantes e investigaré sus logros y sus limitaciones. Estudiaré hasta qué punto los ordenadores actuales pueden hacer cosas que creemos que exigen un alto grado de inteligencia: vencer a los humanos en los juegos más exigentes desde el punto de vista intelectual, traducir de un idioma a otro, responder a preguntas complejas, conducir vehículos por terrenos difíciles. Y examinaré cómo se las arreglan en las cosas que damos por sentadas, las tareas cotidianas que los seres humanos hacemos sin pensar: identificar caras y objetos en imágenes, entender el lenguaje hablado y escrito y utilizar el sentido común más básico.

También intentaré encontrar sentido a las preguntas generales que han dado lugar a los debates sobre la IA desde su nacimiento. ¿A qué nos referimos cuando hablamos de inteligencia «general tan capaz como un humano» o incluso de «inteligencia sobrehumana»? ¿La IA actual está próxima a ese nivel o en camino de alcanzarlo? ¿Qué peligros hay? ¿Qué

aspectos de nuestra inteligencia valoramos más y hasta qué punto una IA tan capaz como un humano pondría en tela de juicio nuestras ideas sobre lo que nos hace humanos? En palabras de Hofstadter, ¿hasta qué punto debemos estar aterrorizados?

Este libro no es un estudio ni una historia general de la inteligencia artificial. Es una exploración detallada de algunos de los métodos de IA que probablemente afectan a nuestra vida —o pronto lo harán— y de los proyectos de IA que más cerca están, tal vez, de cuestionar nuestro sentido de la singularidad humana. Mi objetivo es que los lectores participen de mis indagaciones y que, como yo, intenten hacerse una idea más clara de lo que se ha conseguido en este campo y de lo mucho que queda por hacer antes de que nuestras máquinas puedan defender su propia humanidad.

---

[1] A. Cuthbertson, «DeepMind AlphaGo: AI Teaches Itself ‘Thousands of Years of Human Knowledge’ Without Help», *Newsweek*, 18 de octubre de 2017, www.newsweek.com/deepmind-alphago-ai-teaches-human-help-687620.

[2] Título original: *Gödel, Escher, Bach: an Eternal Golden Braid* [trad. cast. en Barcelona: Booket, 2015].

[3] En las siguientes secciones, las citas de Douglas Hofstadter proceden de una entrevista que mantuve con él después de la reunión de Google; las citas recogen con exactitud el contenido y el tono de sus observaciones ante el grupo de Google.

[4] Jack Schwartz, citado en G.-C. Rota, *Indiscrete Thoughts*, Boston: Berkhäuser, 1997, p. 22.

[5] D. R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, Nueva York: Basic Books, 1979, p. 678.

[6] *Ibid.*, p. 676.

[7] Citado en D. R. Hofstadter, «Staring Emmy Straight in the Eye-and Doing My Best Not to Flinch», en *Creativity, Cognition, and Knowledge*, ed. de T. Dartnell, Westport, Connecticut: Praeger, 2002, pp. 67-100.

[8] Citado en R. Cellan-Jones, «Stephen Hawking Warns Artificial Intelligence Could End Mankind», BBC News, 2 de diciembre de 2014, https://www.bbc.com/news/technology-30290540.

[9] M. McFarland, «Elon Musk: ‘With Artificial Intelligence, We Are Summoning the Demon’», *The Washington Post*, 24 de octubre de 2014.

[10] Bill Gates, en Reddit, 28 de enero de 2015, www.reddit.com/r/IAmA/comments/2tzjp7/hi_redditimbillgatesandimbackformythird/.

[11] Citado en K. Anderson, «Enthusiasts and Skeptics Debate Artificial Intelligence», *Vanity Fair*, 26 de noviembre de 2014.

[12] R. A. Brooks, «Mistaking Performance for Competence», R. A. Books, en *What to Think About Machines That Think*, ed. de J. Brockman, Nueva York: Harper Perennial, 2015, pp. 108-111.

[13] Citado en G. Press, «12 Observations About Artificial Intelligence from the O'Reilly AI Conference», *Forbes*, 31 de octubre de 2016, www.forbes.com/sites/gilpress/2016/10/31/12-observations-about-artificial-intelligence-from-the-oreilly-ai-conference/#886a6012ea2e.

INTELIGENCIA ARTIFICIAL

Guía para **seres pensantes**

A mis padres, que me enseñaron a
ser un ser humano con cabeza
y muchas más cosas

PARTE I

ANTECEDENTES

01

Las raíces de la inteligencia artificial

Dos meses y diez hombres en Dartmouth

El sueño de crear una máquina inteligente —tan inteligente como los humanos o más— nació hace siglos, pero empezó a formar parte de la ciencia moderna con la aparición de los ordenadores digitales. En realidad, las ideas que dieron lugar a los primeros ordenadores programables surgieron de los intentos de los matemáticos de interpretar el pensamiento humano, en especial la lógica, como un proceso mecánico de «manipulación de símbolos». Los ordenadores digitales son esencialmente unos manipuladores de símbolos que juguetean con combinaciones de los símbolos 0 y 1. Varios pioneros de la informática, como Alan Turing y John von Neumann, pensaban que había fuertes analogías entre los ordenadores y el cerebro humano, y les parecía obvio que la inteligencia humana podía reproducirse en programas informáticos.

La mayoría de los expertos en inteligencia artificial atribuyen la fundación oficial de esta materia a un pequeño seminario celebrado en 1956 en el Dartmouth College y organizado por un joven matemático llamado John McCarthy.

En 1955, McCarthy, que tenía veintiocho años, se incorporó a la Facultad de Matemáticas de Dartmouth. Cuando era estudiante había aprendido algo

sobre psicología y el incipiente campo de la «teoría de autómatas» (que más tarde se convertiría en la informática), y le había interesado la idea de crear una máquina pensante. En el Departamento de Matemáticas de Princeton, McCarthy conoció a un compañero, Marvin Minsky, que compartía su fascinación por las posibilidades de los ordenadores inteligentes. Después de graduarse, McCarthy trabajó brevemente en los laboratorios Bell y en IBM, donde colaboró, respectivamente, con Claude Shannon, inventor de la teoría de la información, y con Nathaniel Rochester, un innovador ingeniero eléctrico. Ya en Dartmouth, McCarthy convenció a Minsky, Shannon y Rochester para que le ayudaran a organizar «un estudio de dos meses y diez hombres sobre inteligencia artificial que se llevaría a cabo durante el verano de 1956».[14] El término *inteligencia artificial* lo inventó McCarthy; quería distinguir este campo de otro proyecto relacionado llamado cibernética.[15] Con el tiempo, McCarthy reconoció que el nombre no le gustaba a nadie —al fin y al cabo, el objetivo era una inteligencia genuina, no «artificial»—, pero «tenía que llamarlo de alguna manera, así que lo llamé "inteligencia artificial"».[16]

Los cuatro organizadores presentaron una propuesta a la Fundación Rockefeller y solicitaron que les financiaran el taller estival. El estudio que proponían se basaba en «la hipótesis de que todos los aspectos del aprendizaje o cualquier otra característica de la inteligencia pueden en principio describirse con tanta precisión que es posible hacer que una máquina los imite».[17] La propuesta enumeraba una serie de temas de debate —procesamiento del lenguaje natural, redes neuronales, aprendizaje automático, conceptos abstractos y razonamiento, creatividad— que han seguido definiendo esta materia hasta hoy.

Aunque los ordenadores más avanzados de 1956 eran aproximadamente un millón de veces más lentos que los teléfonos móviles de nuestros días, McCarthy y sus colegas eran optimistas y pensaban que la IA estaba al alcance de la mano: «Creemos que podrá haber avances considerables en

uno o varios de estos problemas si un grupo de científicos minuciosamente escogido se pasa el verano trabajando para conseguirlo».[18]

Pronto surgieron obstáculos que hoy le resultarían familiares a cualquiera que quiera organizar un taller científico. La Fundación Rockefeller solo aportó la mitad del dinero solicitado, y a McCarthy le costó más de lo que creía convencer a los participantes de que acudieran y se quedaran, por no hablar de conseguir que se pusieran de acuerdo en algo. Hubo muchos debates interesantes, pero poca coherencia. Como suele ocurrir en este tipo de reuniones, «cada uno tenía una idea diferente, un sólido ego y mucho entusiasmo por su propio plan».[19] Pese a todo, el verano de la IA en Dartmouth produjo algunos resultados muy importantes. Se dio nombre al campo de investigación y se esbozaron sus objetivos generales. Los que pronto serían los «cuatro grandes» pioneros en la materia —McCarthy, Minsky, Allen Newell y Herbert Simon— se reunieron y empezaron a planear el futuro. Y, por alguna razón, los cuatro salieron de la reunión con enorme optimismo sobre las perspectivas para su campo. A principios de los sesenta, McCarthy fundó el Proyecto de Inteligencia Artificial de Stanford, con «el objetivo de construir una máquina totalmente inteligente antes de que transcurriera una década».[20] Más o menos por aquel entonces, el futuro premio nobel Herbert Simon predijo: «Las máquinas serán capaces, dentro de veinte años, de hacer cualquier tarea que pueda hacer un hombre».[21] Poco después, Marvin Minsky, fundador del Laboratorio de Inteligencia Artificial del MIT, pronosticó que «dentro de una generación […], los problemas de crear una “inteligencia artificial” estarán esencialmente resueltos».[22]

Definiciones y manos a la obra

Ninguna de estas predicciones se ha hecho realidad todavía. Por consiguiente, ¿cuánto nos falta hasta el objetivo de construir una «máquina totalmente inteligente»? ¿Esa máquina necesitaría un trabajo de ingeniería

inversa del cerebro humano en toda su complejidad, o existe un atajo, un conjunto de hábiles algoritmos aún desconocidos, capaz de producir lo que consideramos inteligencia total? ¿Y qué significa «inteligencia total»?

«Define tus términos… o nunca nos entenderemos».[23] Este consejo del filósofo del siglo XVIII Voltaire plantea un reto a cualquiera que hable de inteligencia artificial, porque el concepto central —inteligencia— sigue estando mal definido. Marvin Minsky acuñó la expresión *palabra maleta*[24] para calificar términos del tipo de *inteligencia* y sus muchos parientes, como *pensamiento*, *cognición*, *conciencia* y *emoción*. Cada uno de ellos está lleno como una maleta de un batiburrillo de significados diferentes. La inteligencia artificial ha heredado este problema, puesto que tiene distintos significados en diferentes contextos.

La mayoría de la gente estará de acuerdo en que los humanos son inteligentes y las motas de polvo no. Asimismo, en general creemos que los humanos son más inteligentes que los gusanos. En cuanto a la inteligencia humana, el cociente intelectual se mide en una única escala, pero también hablamos de las distintas dimensiones de la inteligencia: emocional, verbal, espacial, lógica, artística, social, etc. Por consiguiente, la inteligencia puede ser binaria (algo es o no es inteligente), estar en un continuo (una cosa es más inteligente que otra) o ser multidimensional (alguien puede tener mucha inteligencia verbal pero escasa inteligencia emocional). La verdad es que la palabra *inteligencia* es una maleta demasiado llena con la cremallera a punto de romperse.

Para bien o para mal, el ámbito de la IA, en general, ha pasado por alto estas distinciones. En lugar de ello se ha centrado en dos facetas: la científica y la práctica. Desde el punto de vista científico, los investigadores de la IA estudian los mecanismos de la inteligencia «natural» (es decir, biológica) para tratar de integrarla en los ordenadores. Desde el punto de vista práctico, los propulsores de la IA solo quieren crear programas informáticos capaces de llevar a cabo las tareas asignadas tan bien como los

humanos o mejor, sin preocuparse de si esos programas «piensan» verdaderamente como piensan los humanos. Cuando se pregunta a quienes trabajan en IA si sus motivaciones son prácticas o científicas, muchos contestan en broma que depende de quién los financie.

En un informe reciente sobre el estado actual de la IA, un comité de investigadores de renombre definió este campo como «una rama de la informática que estudia las propiedades de la inteligencia a base de sintetizar inteligencia».[25] Un poco en círculo, sí. Pero ese mismo comité reconoció que es difícil definir el campo y que eso puede ser positivo: «La falta de una definición precisa y universalmente aceptada de la IA probablemente ha ayudado a que el campo crezca, se expanda y avance cada vez a más velocidad».[26] Además, señalaba el documento, «los profesionales, investigadores y desarrolladores de la IA actúan guiados por un sentido aproximado del rumbo que deben seguir y por la obligación de “ponerse manos a la obra”».

Una anarquía de enfoques

En el seminario de Dartmouth de 1956, los participantes expresaron diferentes opiniones sobre el enfoque que convenía adoptar para desarrollar la IA. Algunos —sobre todo los matemáticos— propugnaron la lógica matemática y el razonamiento deductivo como lenguaje del pensamiento racional. Otros defendían los métodos inductivos, en los que los programas extraen estadísticas de los datos y utilizan la probabilidad para abordar la incertidumbre. Y otros eran firmes partidarios de inspirarse en la biología y la psicología para crear programas que emularan el cerebro. Lo sorprendente, quizá, es que las discusiones entre los defensores de estos distintos puntos de vista hayan llegado hasta nuestros días. Y cada método ha generado su propia colección de principios y técnicas, reforzada por conferencias y revistas especializadas y con escasa comunicación entre las subespecialidades. Un estudio reciente sobre IA lo resumía así: «Dado que

no comprendemos a fondo la inteligencia ni sabemos cómo producir una IA general, para hacer verdaderos avances, en lugar de cerrar vías de exploración, deberíamos asumir la "anarquía de métodos" de la IA».[27]

Sin embargo, desde la pasada década, hay una familia de métodos de IA —denominada aprendizaje profundo (o redes neuronales profundas)— que ha superado esa anarquía y se ha convertido en el paradigma dominante de la IA. Es más, en muchos medios de comunicación de masas, el término *inteligencia artificial*, en sí, significa hoy «aprendizaje profundo». Es una equivocación desafortunada que nos obliga a aclarar la distinción. La IA es un campo que abarca una gran variedad de enfoques con el propósito de crear máquinas dotadas de inteligencia. El aprendizaje profundo no es más que uno de ellos. En realidad, el aprendizaje profundo es uno de los numerosos métodos existentes en el campo del aprendizaje automático, un subcampo de la IA en el que las máquinas «aprenden» de los datos o de sus propias «experiencias». Para entender mejor estas distinciones, es importante comprender una división filosófica que se produjo en los primeros días de la investigación sobre IA: la división entre las llamadas IA simbólica e IA subsimbólica.

IA simbólica

En primer lugar, veamos la IA simbólica. El conocimiento de un programa de IA simbólica está formado por palabras o frases (los «símbolos»), normalmente comprensibles para un ser humano, además de reglas con arreglo a las cuales el programa combina y procesa esos símbolos para hacer la tarea asignada.

Pondré un ejemplo. Uno de los primeros programas de IA se llamaba nada menos que Solucionador General de Problemas, GPS en sus siglas en inglés.[28] (Lamento la confusión de siglas; este solucionador fue muy anterior al GPS de la geolocalización, o sistema de posicionamiento global). El GPS podía resolver problemas como el de «misioneros y caníbales», con

el que quizá se haya encontrado usted de niño. En este conocido acertijo, tres misioneros y tres caníbales tienen que cruzar un río, pero en su barca solo caben dos personas. Si en algún momento hay más caníbales (hambrientos) que misioneros (de aspecto apetitoso) en una de las dos orillas del río… se pueden imaginar lo que pasa. ¿Cómo consiguen los seis cruzar el río intactos?

Los creadores del Solucionador General de Problemas, los científicos cognitivos Herbert Simon y Allen Newell, grabaron a varios estudiantes mientras «pensaban en voz alta» tratando de resolver este y otros problemas lógicos, y diseñaron su programa para que imitara lo que consideraban los procesos mentales de los estudiantes.

No voy a entrar en detalles sobre cómo funcionaba el GPS, pero se puede ver su carácter simbólico en cómo estaban codificadas las instrucciones del programa. Para plantear el problema, un humano escribiría un código parecido a este:

```
ESTADO ACTUAL:
ORILLA-IZQUIERDA = [3 MISIONEROS, 3 CANÍBALES, 1 BARCA]
ORILLA-DERECHA = [VACÍA]

ESTADO DESEADO:
ORILLA-IZQUIERDA = [VACÍA]
ORILLA-DERECHA = [3 MISIONEROS, 3 CANÍBALES, 1 BARCA]
```

En lenguaje humano, estas líneas representan el hecho de que al principio la orilla izquierda del río «contiene» tres misioneros, tres caníbales y una barca, mientras que la orilla derecha no contiene nada de eso. El estado deseado representa el objetivo del programa: trasladar a todos a la orilla derecha del río.

En cada paso de su procedimiento, el GPS intenta cambiar su estado actual para que se aproxime más al estado deseado. En su código, el programa tiene «operadores» (en forma de subprogramas) capaces de transformar el estado actual en un nuevo estado, así como «reglas» que

codifican los límites de la tarea. Por ejemplo, hay un operador que traslada un número determinado de misioneros y caníbales de un lado a otro del río:

TRASLADAR (#MISIONEROS, #CANÍBALES, DE-LADO A-LADO)

Las palabras dentro de los paréntesis se llaman argumentos, y el programa, cuando está ejecutándose, sustituye esas palabras por números u otras palabras. Es decir, #MISIONEROS se sustituye por el número de misioneros que hay que trasladar, #CANÍBALES por el número de caníbales que hay que trasladar y DE-LADO y A-LADO se sustituyen por «ORILLA-IZQUIERDA» u «ORILLA-DERECHA» dependiendo de la orilla desde la que haya que trasladar a los misioneros y los caníbales. El programa incluye, codificado, el hecho de que el barco se desplaza junto con los misioneros y los caníbales.

Antes de poder aplicar este operador con valores específicos que sustituyan a los argumentos, el programa debe comprobar sus reglas codificadas; por ejemplo, el máximo número de personas que pueden trasladarse a la vez es dos y el operador no se puede usar si el resultado va a ser que haya más caníbales que misioneros en una orilla.

Aunque los símbolos representan conceptos interpretables por un ser humano, como «misioneros», «caníbales», «barco» y «orilla izquierda», el ordenador que ejecuta el programa, por supuesto, no conoce el significado de esos símbolos. Podríamos sustituir «MISIONEROS» por «Z372B» o cualquier otra cadena sin sentido en todos los casos y el programa funcionaría exactamente igual. Esa es una de las cosas a las que se refiere el término *general* en Solucionador General de Problemas. Para el ordenador, el «significado» de los símbolos deriva de las formas en que se pueden combinar, relacionar y manejar.

Los defensores del enfoque simbólico de la IA sostienen que, para que los ordenadores sean inteligentes, no es necesario construir unos programas que imiten el cerebro, sino que es posible capturar totalmente la inteligencia

general con un programa apropiado de procesamiento de símbolos. El funcionamiento de un programa así sería mucho más complejo que el ejemplo de misioneros y caníbales, es cierto, pero seguiría consistiendo en símbolos, combinaciones de símbolos, y reglas y operaciones con símbolos. La IA simbólica del tipo del GPS dominó el campo durante las tres primeras décadas, sobre todo en forma de sistemas expertos, en los que unos expertos humanos concebían reglas para que los programas informáticos las utilizaran en tareas como el diagnóstico médico y la toma de decisiones legales. Hay varias ramas activas de la IA que siguen empleando la IA simbólica; describiré algunos ejemplos más adelante, sobre todo en los debates sobre las aproximaciones de la IA al razonamiento y el sentido común.

IA subsimbólica

La IA simbólica de perceptrones se inspiró originalmente en la lógica matemática y en cómo describía la gente sus procesos de pensamiento consciente. En cambio, los enfoques subsimbólicos de la IA se inspiraban en la neurociencia e intentaban captar los procesos de pensamiento, a veces inconscientes, que sirven de base de lo que algunos denominan percepción rápida, como el reconocimiento facial o la identificación de palabras habladas. Los programas subsimbólicos de IA no contienen un lenguaje comprensible para los humanos como el que vimos en el ejemplo de misioneros y caníbales. Un programa subsimbólico es esencialmente un montón de ecuaciones, una maraña de operaciones numéricas a menudo difíciles de interpretar. Como explicaré enseguida, estos sistemas están diseñados para que, a partir de los datos, aprendan a ejecutar una tarea.

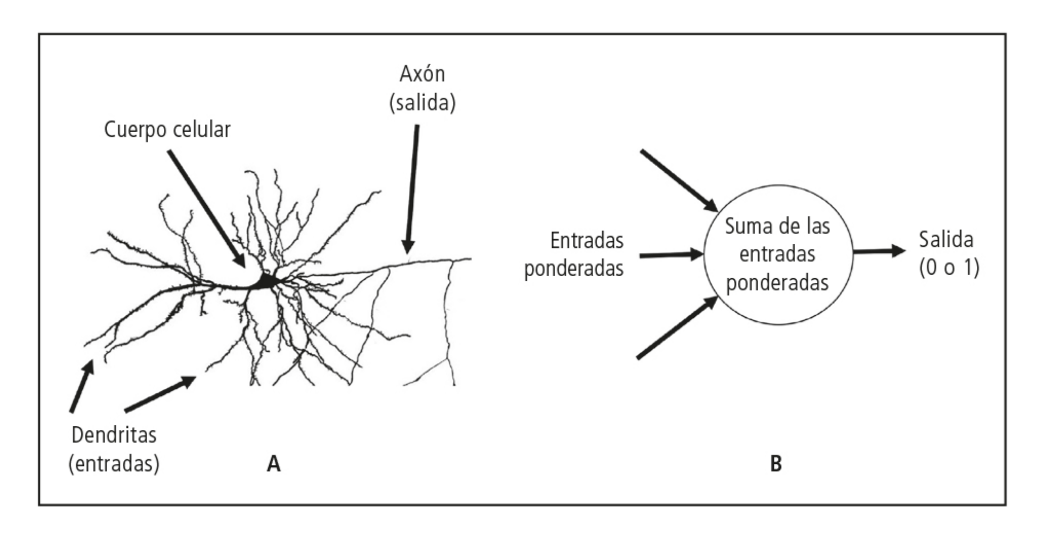


Figura 1. A, una neurona del cerebro; B, un perceptrón simple.

Uno de los primeros ejemplos de programa de IA subsimbólico inspirado en el cerebro fue el perceptrón, inventado a finales de los años cincuenta por el psicólogo Frank Rosenblatt.[29] Para nuestros oídos modernos, el término *perceptrón* puede sonar un poco a la ciencia ficción de aquellos años (como veremos, pronto se le unieron *cognitrón* y *neocognitrón*), pero el perceptrón fue un hito importante en la IA y fue el influyente bisabuelo de la herramienta más eficaz de la IA moderna, las redes neuronales profundas.

Para inventar los perceptrones, Rosenblatt se inspiró en la forma que tienen las neuronas de procesar la información. Una neurona es una célula del cerebro que recibe estímulos eléctricos o químicos de otras neuronas conectadas a ella. Dicho en pocas palabras, una neurona suma todos los datos que recibe de otras neuronas y, si la suma total alcanza un umbral determinado, la neurona se activa. Es importante destacar que las distintas conexiones (sinapsis) de otras neuronas a una neurona concreta tienen distinta potencia; para calcular el total de los datos recibidos, la neurona da más peso a las de las conexiones más fuertes que a las de las más débiles. Los neurocientíficos creen que los ajustes en función de la fuerza de las conexiones entre neuronas son una parte fundamental del aprendizaje en el cerebro.

Para un informático (o, en el caso de Rosenblatt, un psicólogo), el procesamiento de la información en las neuronas puede simularse mediante un programa informático —un perceptrón— con varias entradas numéricas y una salida. En la figura 1 se ilustra la analogía entre una neurona y un perceptrón. La figura 1A muestra una neurona, con sus dendritas ramificadas (las fibras que llevan las informaciones a la célula), el cuerpo celular y el axón (es decir, el canal de salida) etiquetados. La figura 1B muestra un perceptrón simple. El perceptrón, de forma análoga a la neurona, suma sus datos y, si la suma resultante es igual o superior al umbral del perceptrón, este emite el valor uno («se activa»); si no es así, emite el valor cero (no «se activa»). Para simular las distintas fuerzas de las conexiones a una neurona, Rosenblatt propuso que se asignara un peso numérico a cada una de las informaciones que entran en un perceptrón, de forma que cada elemento que entra se multiplica por su peso antes de añadirse a la suma. El umbral de un perceptrón no es más que un número establecido por el programador (o, como veremos, aprendido por el propio perceptrón).

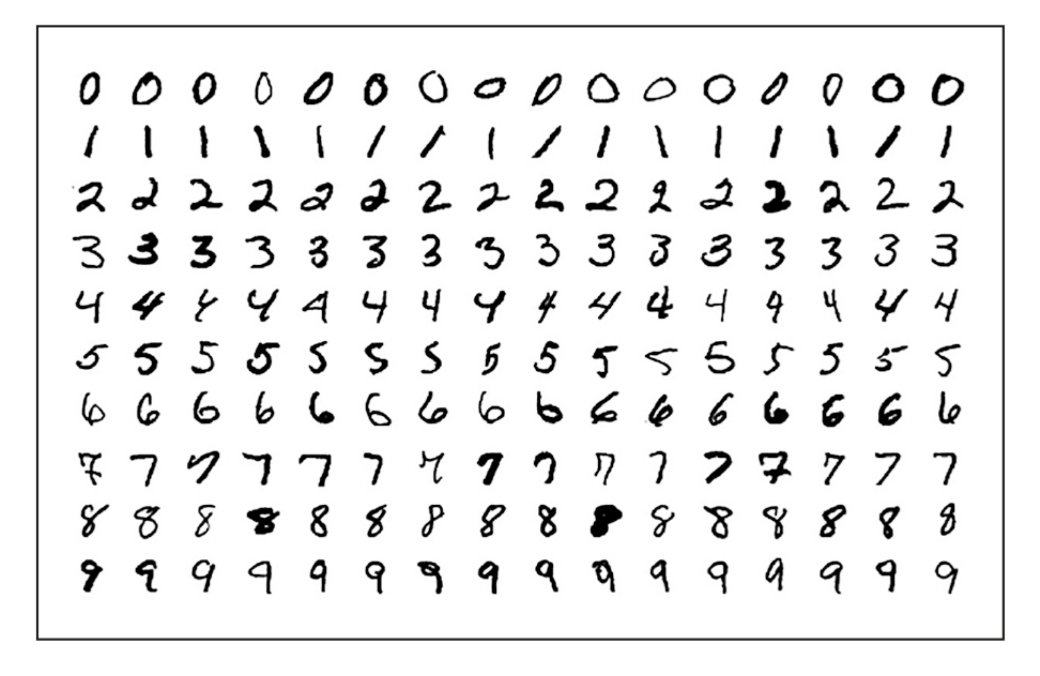


Figura 2. Ejemplos de cifras manuscritas.

En resumen, un perceptrón es un simple programa que decide entre un sí y un no (uno o cero) en función de si la suma de los datos que recibe, ponderados, alcanza un umbral determinado. Probablemente todos tomamos decisiones de este tipo. Por ejemplo, quizá varios amigos nos dicen cuánto les ha gustado una película, pero nos fiamos más del gusto de algunos que del de otros. Si la cantidad total de «entusiasmo de los amigos» —que da más peso a los amigos de los que más confiamos— es suficientemente alta (es decir, superior a algún umbral inconsciente), decidimos ir al cine. Así decidiría un perceptrón ir o no al cine, si tuviera amigos.

Inspirándose en las redes de neuronas del cerebro, Rosenblatt propuso que las redes de perceptrones pudieran ejecutar tareas visuales como el reconocimiento de caras y objetos. Para hacernos una idea de cómo podrían hacerlo, vamos a ver cómo se podría usar un perceptrón para una tarea visual concreta: reconocer cifras manuscritas como las de la figura 2.

En concreto, vamos a diseñar un perceptrón que sea detector de ochos, es decir, que emita un uno si los datos que recibe proceden de una imagen que representa un ocho, y que emita un cero si la imagen representa alguna otra cifra. Para diseñar un detector de este tipo hay que (1) averiguar cómo convertir una imagen en un conjunto de informaciones numéricas y (2) determinar los números que se van a usar para la ponderación y el umbral del perceptrón, de modo que dé la emisión correcta (uno en el caso de ocho; cero en el caso de otras cifras). Voy a explicar algunos detalles al respecto porque más adelante, cuando hable sobre redes neuronales y sus aplicaciones en visión artificial, volverán a aparecer muchas de estas mismas ideas.

Las informaciones que entran en nuestro perceptrón

La figura 3A muestra un ocho manuscrito ampliado. Cada elemento de la cuadrícula es un píxel con un valor numérico de «intensidad»: los

cuadrados blancos tienen una intensidad de cero, los negros de uno y los grises están entre los dos. Supongamos que las imágenes que damos a nuestro perceptrón se han ajustado para que tengan el mismo tamaño que esta: 18 × 18 píxeles. La figura 3B ilustra un perceptrón diseñado para reconocer los ochos. Este perceptrón tiene 324 entradas (es decir, 18 × 18), cada una de las cuales corresponde a uno de los píxeles de la cuadrícula de 18 × × 18 píxeles. En una imagen como la de la figura 3A, cada entrada del perceptrón se ajusta a la intensidad del píxel correspondiente. Cada entrada tendría su propio valor de ponderación (no mostrado en la figura).

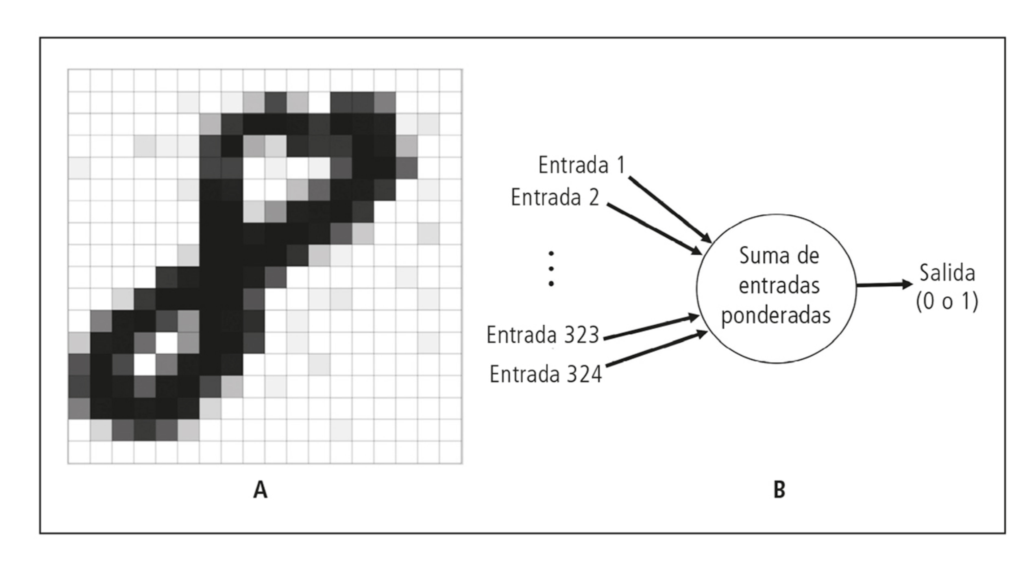


Figura 3. Ilustración de un perceptrón que reconoce un ocho manuscrito. Cada píxel de la imagen de 18 × 18 píxeles corresponde a una entrada de información del perceptrón, de modo que hay 324 (= 18 × 18) entradas.

Aprendizaje de los pesos y umbrales del perceptrón

A diferencia del Solucionador General de Problemas, el sistema simbólico que describí más arriba, un perceptrón no tiene unas reglas explícitas para ejecutar su tarea; todo su «conocimiento» está codificado en los números que constituyen sus pesos y umbrales. En sus diversos ensayos, Rosenblatt demostró que, dados los valores correctos de peso y umbral, un perceptrón

como el de la figura 3B puede llevar a cabo bastante bien tareas de percepción como el reconocimiento de cifras manuscritas sencillas. Ahora bien, ¿cómo determinar exactamente los pesos y umbrales correctos para una tarea determinada? También aquí, Rosenblatt respondió tomando ejemplo del cerebro: el perceptrón debería aprender esos valores por sí solo. ¿Y cómo se supone que va a aprender los apropiados? En consonancia con las teorías de psicología conductista populares en la época, la idea de Rosenblatt era que los perceptrones debían aprender mediante el condicionamiento. Inspirándose en parte en el psicólogo conductista B. F. Skinner, que enseñaba a ratas y palomas a llevar a cabo diversas tareas a base de refuerzos positivos y negativos, Rosenblatt pensó que el perceptrón debía entrenarse a partir de ejemplos: con una recompensa cuando «se activase» bien y con un castigo cuando se equivocase. Este condicionamiento es el que hoy se conoce en el campo de la IA como aprendizaje supervisado. Durante el entrenamiento, el sistema de aprendizaje recibe un ejemplo, emite una salida y entonces recibe una «señal de supervisión», que indica hasta qué punto difiere lo emitido por el sistema del resultado correcto, de manera que el sistema utiliza esa señal para ajustar sus pesos y umbrales.

El concepto de aprendizaje supervisado es parte fundamental de la IA moderna, así que merece la pena que nos detengamos en él. Normalmente, el aprendizaje supervisado necesita una gran cantidad de ejemplos positivos (por ejemplo, una colección de números ocho escritos por distintas personas) y negativos (por ejemplo, una colección de cifras distintas de ocho, también manuscritas). Un humano etiqueta cada ejemplo según su categoría: en este caso, ocho o no ocho. Esta etiqueta va a ser la señal de supervisión. Para entrenar al sistema se emplean varios de los ejemplos positivos y negativos, lo que se denomina «datos de entrenamiento». Los demás, los «datos de prueba», se usan para evaluar el rendimiento del sistema después de entrenarlo, para ver hasta qué punto ha aprendido a dar

la respuesta correcta en general, no solo en los ejemplos con los que se le ha entrenado.

El término más importante en informática es tal vez *algoritmo*, que designa una «receta» con los pasos que puede seguir un ordenador para resolver un problema concreto. La principal contribución de Frank Rosenblatt a la IA fue el diseño de un algoritmo específico, llamado algoritmo de aprendizaje del perceptrón, con el que es posible entrenar un perceptrón a partir de ejemplos para determinar las ponderaciones y el umbral necesarios para emitir respuestas correctas. Funciona así: para empezar, se asignan a los pesos y el umbral unos valores aleatorios entre −1 y 1. En nuestro ejemplo, la ponderación de la primera entrada de información podría fijarse en 0,2, la de la segunda entrada en −0,6, y así sucesivamente, mientras que el umbral podría ser 0,7. Un programa informático llamado generador de números aleatorios puede generar cómodamente estos valores iniciales.

Ahora podemos iniciar el proceso de entrenamiento. Se da al perceptrón el primer ejemplo de entrenamiento, sin que vea todavía la etiqueta de categoría correcta. El perceptrón multiplica cada entrada por su peso, suma todos los resultados, compara la suma con el umbral y emite un uno o un cero. En este caso, el uno significa una conjetura de ocho y el cero significa una conjetura de no ocho. Entonces, el proceso de entrenamiento compara lo emitido por el perceptrón con la respuesta correcta que aparece en la etiqueta asignada por el ser humano (es decir, ocho o no ocho). Si el perceptrón acierta, los pesos y el umbral no cambian. Pero si el perceptrón se equivoca, se modifican ligeramente, para que la suma del perceptrón en este ejemplo de entrenamiento se aproxime más a los valores que producen la respuesta correcta. Además, el grado de modificación de cada peso depende del valor de entrada asociado; es decir, la responsabilidad del error se asigna en función de las entradas o estímulos que hayan tenido más efecto. Por ejemplo, en el ocho de la figura 3A, los píxeles de mayor

intensidad (en este caso, negros) tienen más efecto, y los píxeles con intensidad cero (en este caso, blancos) no tendrían ningún impacto. (Para los lectores interesados, he incluido algunos detalles matemáticos en las notas).[30]

Todo el proceso se repite con el siguiente ejemplo de entrenamiento. Se utilizan todos los ejemplos de entrenamiento varias veces y se modifican ligeramente los pesos y el umbral cada vez que el perceptrón comete un error. Como descubrió el psicólogo B. F. Skinner cuando entrenaba palomas, es mejor aprender de forma gradual, probando muchas veces; si se modifican los pesos y el umbral demasiado de una sola vez, el sistema puede acabar aprendiendo lo que no debe (como la generalización de que «las mitades inferior y superior de un ocho tienen siempre el mismo tamaño»). Después de muchas repeticiones con cada ejemplo de entrenamiento, el sistema acaba (si todo va bien) por establecer un conjunto de pesos y un umbral que dan como resultado respuestas correctas para todos los ejemplos de entrenamiento. Entonces podemos evaluar el perceptrón con los ejemplos de prueba para ver cómo funciona con imágenes para las que no ha sido entrenado. Un detector de ochos es útil si nos interesan únicamente los ochos. Pero ¿y si queremos que reconozca otras cifras? Es bastante fácil ampliar nuestro perceptrón para que tenga diez salidas, una por cada dígito. Dada una cifra manuscrita de ejemplo, la salida correspondiente a esa cifra debe ser uno y todas las demás salidas deben ser cero. Este perceptrón ampliado puede aprender todos los pesos y umbrales utilizando el algoritmo de aprendizaje del perceptrón; lo único que necesita el sistema son suficientes ejemplos.

Rosenblatt y otros demostraron que las redes de perceptrones podían aprender a desempeñar tareas perceptivas relativamente sencillas; además, Rosenblatt demostró matemáticamente que, para una clase de tareas determinada, aunque muy concreta, los perceptrones suficientemente entrenados podían, en principio, aprender a ejecutar esas tareas sin errores.

Lo que no estaba claro era hasta qué punto los perceptrones podían hacer bien tareas de IA más generales, pero esa incertidumbre no pareció impedir que Rosenblatt y sus patrocinadores de la Oficina de Investigaciones Navales hicieran predicciones absurdamente optimistas sobre su algoritmo. *The New York Times*, en su información sobre una rueda de prensa que ofreció Rosenblatt en julio de 1958, hizo este resumen:

> La Armada ha dado a conocer hoy el embrión de un ordenador electrónico que prevé que podrá caminar, hablar, ver, escribir, reproducirse y ser consciente de su propia existencia. Con el tiempo, pronostican, los perceptrones serán capaces de reconocer a una persona y llamarla por su nombre, así como de traducir al instante un idioma hablado a otro hablado y escrito.[31]

Sí, ya desde el principio, la IA fue víctima de un exceso de expectativas. Enseguida diré algo más sobre los desgraciados resultados de esta exageración. De momento, quiero utilizar los perceptrones para destacar una diferencia importante entre el enfoque simbólico de la IA y el subsimbólico.

El hecho de que los «conocimientos» de un perceptrón consistan en una serie de números —en concreto, los pesos y el umbral que ha aprendido— hace que sea difícil descubrir las reglas que utiliza para llevar a cabo su tarea de reconocimiento. Las reglas del perceptrón no son simbólicas; a diferencia de los símbolos del Solucionador General de Problemas, como ORILLA-IZQUIERDA, #MISIONEROS Y TRASLADAR, los pesos y el umbral de un perceptrón no representan conceptos concretos. No es fácil traducir esos números en reglas comprensibles para los humanos. Y la situación es mucho peor con las redes neuronales modernas, que tienen millones de pesos.

Podríamos hacer una vaga analogía entre los perceptrones y el cerebro humano. Si pudiéramos abrirnos la cabeza y observar cómo funciona un subconjunto de los cientos de miles de millones de neuronas que contiene, probablemente no entenderíamos lo que estamos pensando ni las «reglas» que nos han servido para tomar una decisión concreta. Sin embargo, el

cerebro humano ha creado el lenguaje, que permite utilizar símbolos (palabras y frases) para contarnos —muchas veces de forma imperfecta— qué está pensando una persona o por qué ha hecho tal cosa. En este sentido, nuestras activaciones neuronales pueden considerarse «subsimbólicas», porque sustentan los símbolos que de una u otra manera crea nuestro cerebro. Los perceptrones y otras redes de neuronas simuladas más complicadas se denominan «subsimbólicos» por analogía con el cerebro. Sus defensores creen que, para conseguir la inteligencia artificial, los símbolos equivalentes al lenguaje y las reglas que rigen el procesamiento de símbolos no pueden programarse directamente, como se hizo en el Solucionador General de Problemas, sino que deben surgir de arquitecturas similares a las neuronales, de la misma forma que el procesamiento inteligente de símbolos surge del cerebro.

Las limitaciones de los perceptrones

Después de la reunión de Dartmouth de 1956, el campo de la IA estuvo dominado por el bando simbólico. En los primeros años sesenta, mientras Rosenblatt se dedicaba ávidamente a desarrollar el perceptrón, los cuatro grandes «fundadores» de la IA, todos ellos firmes partidarios de ese enfoque, habían creado ya laboratorios de inteligencia artificial bien dotados e influyentes: Marvin Minsky en el MIT, John McCarthy en Stanford y Herbert Simon y Allen Newell en Carnegie Mellon. (Es interesante que estas tres universidades sigan siendo hoy tres de los lugares más prestigiosos para estudiar IA). Minsky, en concreto, pensaba que la teoría de Rosenblatt sobre la IA, inspirada en el cerebro, era un callejón sin salida que estaba quitando dinero a las investigaciones de tipo simbólico, que merecían más la pena.[32] En 1969, Minsky y su colega del MIT Seymour Papert publicaron un libro, *Perceptrons*,[33] en el que probaban matemáticamente que los tipos de problemas que podía resolver de forma

perfecta un perceptrón eran muy limitados y que su algoritmo de aprendizaje no serviría para tareas que exigiesen muchos pesos y umbrales.

Minsky y Papert señalaron que si se amplificaba un perceptrón con el añadido de una «capa» de neuronas simuladas, podría resolver, en principio, muchos más tipos de problemas.[34] Un perceptrón al que se ha añadido una capa de este tipo se denomina red neuronal multicapa. Estas redes son la base de gran parte de la IA moderna; las describiré con detalle en el próximo capítulo. Por ahora, me limitaré a indicar que en la época del libro de Minsky y Papert las redes neuronales multicapa no eran objeto de muchos estudios, sobre todo porque no había un algoritmo general análogo al algoritmo de aprendizaje del perceptrón para memorizar pesos y umbrales.

Las limitaciones de los perceptrones simples que demostraron Minsky y Papert ya eran conocidas por los especialistas de este campo.[35] El propio Frank Rosenblatt había investigado mucho sobre los perceptrones multicapa y era consciente de las dificultades de entrenarlos.[36] Lo que acabó definitivamente con el perceptrón no fueron los datos matemáticos de Minsky y Papert, sino sus especulaciones sobre las redes neuronales multicapa:

> [El perceptrón] tiene muchas características que llaman la atención: su linealidad; su curioso teorema de aprendizaje; su clara simplicidad paradigmática como forma de computación paralela. No hay nada que indique que esas virtudes pueden trasladarse a la versión multicapa. Aun así, creemos que un problema de investigación importante es esclarecer (o rechazar) nuestra opinión intuitiva de que la ampliación es estéril.[37]

Ay. En la jerga actual, se podría decir que esa frase final era «pasivo-agresiva». Las especulaciones negativas de este tipo fueron, al menos en parte, el motivo de que las investigaciones sobre redes neuronales se quedaran sin financiación a finales de los años sesenta, mientras que la IA simbólica empezó a recibir montañas de dinero público. En 1971, cuando tenía cuarenta y tres años, Frank Rosenblatt murió en un accidente náutico. Sin su principal impulsor y con escasos fondos públicos, las investigaciones

sobre los perceptrones y otros métodos subsimbólicos de IA se interrumpieron casi por completo, excepto entre algunos grupos académicos aislados.

El invierno de la IA

Mientras tanto, los defensores de la IA simbólica se dedicaban a redactar propuestas de subvenciones que prometían avances inminentes en ámbitos como la comprensión del habla y el lenguaje, el razonamiento basado en el sentido común, la navegación robótica y los vehículos autónomos. A mediados de los años setenta, si bien habían conseguido poner en marcha algunos sistemas expertos con objetivos muy concretos, los avances generales de la IA prometidos no se habían hecho realidad.

Las entidades de financiación se dieron cuenta. Dos informes, solicitados respectivamente por el Consejo de Investigaciones Científicas del Reino Unido y el Departamento de Defensa de Estados Unidos, presentaron conclusiones muy negativas sobre los avances y las perspectivas de la investigación en IA. En concreto, el informe británico reconocía que había señales prometedoras en el área de los sistemas expertos especializados —«programas escritos para ámbitos de problemas muy especializados, en los que la programación tiene muy en cuenta las lecciones de la experiencia y la inteligencia humana en el ámbito en cuestión»—, pero su conclusión era que los resultados hasta la fecha eran «totalmente desalentadores en los programas de ámbito general, que tratan de emular los aspectos de la actividad [cerebral] humana dedicados a la resolución de problemas en un campo muy amplio. Crear un programa general de este tipo, el objetivo a largo plazo al que aspiran las investigaciones sobre IA, parece estar tan lejos como siempre».[38] En el Reino Unido, este informe provocó que disminuyeran a toda velocidad los fondos públicos destinados a las investigaciones sobre IA, de la misma manera que, en Estados Unidos, el

Departamento de Defensa recortó drásticamente las subvenciones para la investigación básica sobre IA.

Este fue uno de los primeros ejemplos de un ciclo constante de burbujas y estallidos en el campo de la IA. El ciclo tiene dos partes. Fase 1: aparecen nuevas ideas que inspiran gran optimismo en la comunidad investigadora. Se prometen —muchas veces con gran revuelo mediático— avances inminentes en el campo de la IA. Las administraciones públicas y los inversores de capital riesgo aportan fondos para la investigación académica y para empresas privadas emergentes. Fase 2: los avances prometidos no se materializan o son mucho menos impresionantes de lo que se esperaba. La financiación pública y el capital riesgo se agotan. Las empresas emergentes desaparecen y la investigación sobre IA empieza a ir más despacio. Este patrón se ha vuelto familiar para el mundo de la IA: a la «primavera de la IA» le siguen todo tipo de promesas y el despliegue en los medios, y después, el «invierno de la IA». Esto ha sucedido, con algunas variaciones, en ciclos de cinco a diez años. Cuando terminé mis estudios de posgrado en 1990, el campo de la IA pasaba por uno de sus inviernos y había adquirido tan mala imagen que incluso me aconsejaron que dejara de utilizar el término *inteligencia artificial* cuando solicitara empleo.

Las cosas fáciles son difíciles

Los fríos inviernos de la IA enseñaron varias lecciones importantes a los profesionales. La más sencilla fue la que subrayó John McCarthy cincuenta años después de la conferencia de Dartmouth: «La IA era más difícil de lo que pensábamos».[39] Marvin Minsky señaló que, en realidad, las investigaciones sobre IA habían puesto al descubierto una paradoja: «Las cosas fáciles son difíciles». Los objetivos originales de la IA —unos ordenadores capaces de conversar con nosotros en lenguaje natural, describir lo que ven a través de los ojos de su cámara, aprender nuevos conceptos con solo ver unos cuantos ejemplos— son cosas que cualquier

niño pequeño puede hacer con facilidad, pero lo sorprendente es que para la IA es más difícil llevar a cabo esas «cosas fáciles» que diagnosticar enfermedades complejas, vencer a campeones humanos de ajedrez y go y resolver complejos problemas algebraicos. Como dijo también Minsky, «en general, somos menos conscientes de lo que nuestras mentes hacen mejor».[40] Por lo menos, el intento de crear una inteligencia artificial ha ayudado a dilucidar lo compleja y sutil que es nuestra propia mente.

---

[14] J. McCarthy *et al.*, «A Proposal for the Dartmouth Summer Research Project in Artificial Intelligence», presentado a la Fundación Rockefeller, 1955, reimpreso en *AI Magazine* 27, n.º 4 (2006), pp. 12-14.

[15] La cibernética era un campo interdisciplinario que estudiaba «el control y la comunicación en el animal y en las máquinas». Véase N. Wiener, *Cybernetics*, Cambridge, Mass.: MIT Press, 1961.

[16] Citado en N. J. Nilsson, *John McCarthy: A Biographical Memoir,* Washington D. C.: National Academy of Sciences, 2012.

[17] McCarthy *et al.*, «Proposal for the Dartmouth Summer Research Project in Artificial Intelligence».

[18] *Ibid.*

[19] G. Solomonoff, «Ray Solomonoff and the Dartmouth Summer Research Project in Artificial Intelligence, 1956», consultado el 4 de diciembre de 2018, www.raysolomonoff.com/dartmouth/dartray.pdf.

[20] H. Moravec, *Mind Children: The Future of Robot and Human Intelligence*, Cambridge, Mass.: Harvard University Press, 1988, p. 20.

[21] H. A. Simon, *The Shape of Automation for Men and Management,* Nueva York: Harper & Row, 1965, p. 96. Obsérvese que el uso por Simon de «hombre» en lugar de «persona» era habitual en los Estados Unidos de los años sesenta.

[22] M. L. Minsky, *Computation: Finite and Infinite Machines*, Upper Saddle River, N. J.: Prentice-Hall, 1967, p. 2.

[23] B. R. Redman, *The Portable Voltaire,* Nueva York: Penguin Books, 1977, p. 225.

[24] M. L. Minsky, *The Emotion Machine: Commonsense Thinking, Artificial Intelligence, and the Future of the Human Mind,* Nueva York: Simon & Schuster, 2006, p. 95 [trad. cast.: *La máquina de las emociones. Sentido común, inteligencia artificial y el futuro de la mente humana*, Barcelona: Editorial Debate, 2010].

[25] *One Hundred Year Study on Artificial Intelligence (AI100)*, «2016 Report», p. 13, ai100.stanford.edu/2016-report.

[26] *Ibid.*, p. 12.

[27] J. Lehman, J. Clune y S. Risi, «An Anarchy of Methods: Current Trends in How Intelligence Is Abstracted in AI», *IEEE Intelligent Systems* 29, n.º 6 (2014), pp. 56-62.

[28] A. Newell y H. A. Simon, «GPS: A Program That Simulates Human Thought», P-2257, Rand Corporation, Santa Monica, Calif., 1961.

[29] F. Rosenblatt, «The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain», *Psychological Review* 65, n.º 6 (1958), pp. 386-408.

[30] Matemáticamente, el algoritmo de aprendizaje del perceptrón es el siguiente. Para cada peso w_j: $w_j \leftarrow w_j + \eta (t - y) x_j$, donde *t* es la salida correcta (1 o 0) para la entrada dada, *y* es la salida real del perceptrón, x_j es la entrada asociada al peso w_j, y η es la *tasa de aprendizaje*, un valor dado por el programador. La flecha significa una actualización. El umbral se incorpora mediante la creación de una «entrada» adicional x_0 con un valor constante de 1, cuyo peso asociado es $w_0 = -umbral$. Con esta entrada y este peso (llamado sesgo), el perceptrón se dispara solo si la suma de las entradas multiplicadas por los pesos (es decir, el producto punto entre el vector de entrada y el vector de pesos) es mayor o igual que cero. A menudo, los valores de entrada se escalan y se aplican otras transformaciones para evitar que los pesos crezcan demasiado.

[31] Citado en M. Olazaran, «A Sociological Study of the Official History of the Perceptrons Controversy», *Social Studies of Science* 26, n.º 3 (1996), pp. 611-659.

[32] M. A. Boden, *Mind as Machine: A History of Cognitive Science*, Oxford: Oxford University Press, 2006, 2:913.

[33] M. L. Minsky y S. L. Papert, *Perceptrons: An Introduction to Computational Geometry*, Cambridge, Mass.: MIT Press, 1969.

[34] En términos técnicos, cualquier función booleana puede calcularse mediante una red multicapa totalmente conectada con unidades lineales de umbral y una capa interna («oculta»).

[35] Olazaran, «Sociological Study of the Official History of the Perceptrons Controversy».

[36] G. Nagy, «Neural Networks—Then and Now», *IEEE Transactions on Neural Networks* 2, n.º 2 (1991), pp. 316-318.

[37] Minsky y Papert, *Perceptrons*, pp. 231-232.

[38] J. Lighthill, «Artificial Intelligence: A General Survey», en *Artificial Intelligence: A Paper Symposium*, Londres: Science Research Council, 1973.

[39] Citado en C. Moewes y A. Nürnberger, *Computational Intelligence in Intelligent Data Analysis*, Nueva York: Springer, 2013, p. 135.

[40] M. L. Minsky, *The Society of Mind*, Nueva York: Simon & Schuster, 1987, p. 29 [trad. cast.: *La sociedad de la mente*, Buenos Aires: Ediciones Galápago, 1986].

02

Las redes neuronales y el auge del aprendizaje automático

Alerta de *spoiler*: las redes neuronales multicapa —la amplificación de los perceptrones que Minsky y Papert descartaron porque la consideraban «estéril»— han acabado siendo el punto de partida de gran parte de la inteligencia artificial moderna. Dado que en esas redes se basan varios de los métodos que describiré en capítulos posteriores, dedicaré algo de tiempo a describir cómo funcionan.

Redes neuronales multicapa

Una red no es más que un conjunto de elementos conectados entre sí de diversas maneras. Todos sabemos lo que son las redes sociales, en las que los elementos son personas, y las redes informáticas, en las que los elementos son, por supuesto, ordenadores. En las redes neuronales, los elementos son neuronas simuladas parecidas a los perceptrones que describí en el capítulo anterior.

En la figura 4 he esbozado una sencilla red neuronal multicapa diseñada para reconocer cifras escritas a mano. La red tiene dos columnas (capas) de neuronas simuladas tipo perceptrón (círculos). Para simplificar (y probablemente para alivio de los neurocientíficos que me lean), voy a

describir los elementos de esta red empleando el término *unidad*, en vez de *neurona*. La red de la figura 4, como el perceptrón detector de ochos del capítulo 1, tiene 324 (18 × 18) entradas (estímulos), cada una de ellas ajustada al valor de intensidad del píxel correspondiente en la imagen de entrada. Sin embargo, a diferencia del perceptrón, esta red tiene una capa de tres unidades denominadas ocultas, además de su capa de diez unidades de salida. Cada unidad de salida corresponde a una de las categorías de cifras posibles.

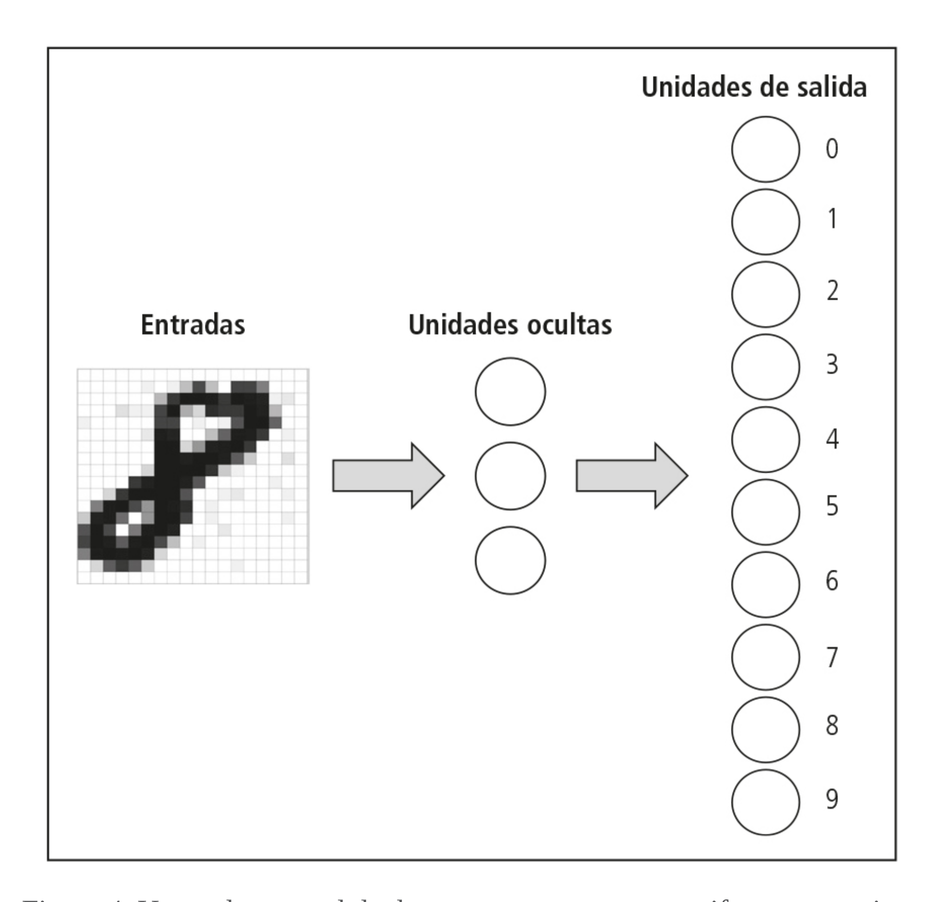


Figura 4. Una red neuronal de dos capas para reconocer cifras manuscritas.

Las grandes flechas grises indican que cada estímulo tiene una conexión ponderada con cada unidad oculta y que cada unidad oculta tiene una conexión ponderada con cada unidad de salida. El término *unidad oculta*, de aire tan misterioso, procede de los textos sobre redes neuronales y se

refiere sencillamente a una unidad que no es de salida. Quizá habría sido más apropiado llamarla «unidad interior».

Pensemos en la estructura de nuestro cerebro, en el que algunas neuronas controlan directamente «salidas» como los movimientos musculares, pero las demás, en su mayoría, se limitan a comunicarse con otras neuronas. Estas son las que podríamos llamar las neuronas ocultas del cerebro.

La red de la figura 4 se denomina «multicapa» porque tiene dos capas de unidades (ocultas y de salida), en vez de una capa de salida y nada más. En principio, una red multicapa puede tener varias capas de unidades ocultas; las redes que tienen más de una capa de unidades ocultas se llaman redes profundas. La «profundidad» de una red no es más que el número de capas ocultas que contiene. En los próximos capítulos hablaré mucho más de las redes profundas.

Igual que sucede en los perceptrones, cada unidad multiplica cada una de sus entradas por el peso de la conexión de esa entrada y suma los resultados. Sin embargo, a diferencia de lo que pasa en un perceptrón, aquí una unidad no se limita a «activarse» o «no activarse» (es decir, emitir uno o cero) en función de un umbral, sino que cada unidad utiliza su suma para calcular un número entre cero y uno que es la llamada «activación» de la unidad. Si la suma que calcula una unidad es baja, la activación de la unidad se aproxima a cero; si la suma es alta, la activación se aproxima a uno. (Para los lectores interesados, he incluido algunos de los detalles matemáticos en las notas).[41]

Para procesar una imagen como el ocho manuscrito de la figura 4, la red lleva a cabo sus cálculos capa a capa, de izquierda a derecha. Cada unidad oculta calcula su valor de activación; esos valores de activación se convierten en las entradas para las unidades de salida, que, a su vez, calculan sus propias activaciones. En la red de la figura 4, se puede considerar que la activación de una unidad de salida es la confianza de la

red en que está «viendo» la cifra correspondiente; la categoría de cifras que tenga más confianza es la respuesta de la red, su clasificación.

En teoría, una red neuronal multicapa puede aprender a utilizar sus unidades ocultas para reconocer características más abstractas (por ejemplo, formas como los «círculos» superior e inferior de un ocho escrito a mano) que las básicas (por ejemplo, píxeles) codificadas en la entrada. No suele ser fácil saber de antemano cuántas capas de unidades ocultas se necesitan ni cuántas unidades ocultas deben incluirse en una capa para que una red desempeñe bien una tarea determinada. Los investigadores de redes neuronales recurren en general a alguna forma de prueba y error para descubrir los ajustes más apropiados.

Aprendizaje por retropropagación

En su libro *Perceptrons*, Minsky y Papert se mostraban escépticos ante la posibilidad de diseñar un algoritmo capaz de aprender los pesos de una red neuronal multicapa. Su escepticismo, unido a las dudas de otros estudiosos de la IA simbólica, fue una de las principales razones de que disminuyeran bruscamente los fondos para la investigación sobre redes neuronales en los años setenta. Sin embargo, a pesar del jarro de agua fría que supuso el libro de Minsky y Papert, un pequeño grupo de investigadores sobre redes neuronales persistió, sobre todo en el campo de la psicología cognitiva de Frank Rosenblatt, de forma que a finales de los años setenta y principios de los ochenta, varios de ellos habían conseguido ya refutar de forma definitiva las especulaciones de Minsky y Papert sobre la «esterilidad» de las redes neuronales multicapa mediante el desarrollo de un algoritmo de aprendizaje general —llamado retropropagación— para entrenar estas redes.

Como su nombre indica, la retropropagación consiste en fijarse en un error observado en las unidades de salida (por ejemplo, la alta confianza con respecto a una cifra errónea en la figura 4) y «propagar» la culpa de ese

error hacia atrás (en la figura 4, sería de derecha a izquierda) para asignar la responsabilidad correspondiente a cada uno de los pesos de la red. Eso permite que la retropropagación determine cuánto hay que modificar cada peso para disminuir el error. El aprendizaje en redes neuronales consiste sencillamente en modificar de forma gradual los pesos de las conexiones para que cada error de salida se acerque lo más posible a cero en todos los ejemplos de entrenamiento. Aunque aquí no voy a abordar los aspectos matemáticos de la retropropagación, he incluido algunos detalles en las notas.[42]

La retropropagación funciona (al menos en teoría) independientemente del número de entradas, unidades ocultas o unidades de salida que tenga la red neuronal. Aunque no existen garantías matemáticas de que la retropropagación resuelva los pesos correctos para una red, en la práctica ha conseguido buenos resultados en muchas tareas que son demasiado difíciles para los perceptrones simples. Por ejemplo, yo entrené un perceptrón y una red neuronal de dos capas, cada una con trescientas veinticuatro entradas y diez salidas, para que aprendieran a reconocer cifras escritas a mano; utilicé sesenta mil ejemplos y luego examiné hasta qué punto era capaz cada uno de ellos de reconocer diez mil ejemplos nuevos. El perceptrón acertó en el 80 por ciento de los nuevos ejemplos, mientras que la red neuronal, con cincuenta unidades ocultas, acertó en el 94 por ciento de ellos. Bien hecho, unidades ocultas. Pero ¿qué había aprendido exactamente la red neuronal para superar al perceptrón? No lo sé. A lo mejor encuentro alguna forma de visualizar las 16.700 ponderaciones de la red neuronal[43] para comprender mejor su rendimiento, pero de momento no lo he conseguido y, en general, no es nada fácil entender cómo toman sus decisiones estas redes.

Es importante subrayar que, aunque he utilizado el ejemplo de las cifras manuscritas, las redes neuronales no solo pueden aplicarse a imágenes, sino a cualquier tipo de datos. Las redes neuronales se han utilizado en ámbitos

tan distintos como el reconocimiento de voz, las predicciones en el mercado bursátil, la traducción y la composición musical.

Conexionismo

En los años ochenta, el grupo más conocido del campo de las redes neuronales era un equipo de la Universidad de California en San Diego dirigido por dos psicólogos, David Rumelhart y James McClelland. En aquel entonces, las que hoy llamamos redes neuronales solían denominarse redes conexionistas, un término que expresa la idea de que, en estas redes, el conocimiento reside en las conexiones ponderadas entre unidades. El equipo que dirigían Rumelhart y McClelland es conocido porque escribió la llamada biblia del conexionismo: un tratado en dos volúmenes, publicado en 1986, titulado *Parallel Distributed Processing*. En un panorama dominado por la IA simbólica, el libro era un alegato en favor del enfoque subsimbólico, que sostenía que «las personas son más inteligentes que los ordenadores actuales porque el cerebro emplea una arquitectura computacional básica más adecuada para —por ejemplo— percibir objetos en escenarios naturales e identificar sus relaciones, […] comprender el lenguaje y recuperar información contextualmente apropiada de la memoria».[44] Los autores especulaban con que «los sistemas simbólicos como los que prefieren Minsky y Papert»[45] no serían capaces de captar esas capacidades humanas.

De hecho, a mediados de los años ochenta, los sistemas expertos —unos enfoques de IA simbólica que se basan en los humanos para crear reglas que reflejen el conocimiento experto de un ámbito concreto— empezaban a parecer cada vez más frágiles: es decir, propensos a errores y a menudo incapaces de generalizar o adaptarse a situaciones nuevas. Al analizar las limitaciones de estos sistemas, los investigadores descubrieron hasta qué punto los expertos humanos que escriben las reglas se basan, en realidad, en conocimientos subconscientes —lo que podríamos llamar sentido común—

para actuar de forma inteligente. Este tipo de sentido común no era fácil de recoger en reglas programadas ni deducciones lógicas, y su falta limitaba gravemente cualquier aplicación general de los métodos de IA simbólica. Es decir, después de un ciclo de grandes promesas, mucho dinero y revuelo mediático, la IA simbólica afrontaba otro invierno.

Los partidarios del conexionismo decían que las claves de la inteligencia eran una arquitectura computacional adecuada —inspirada en el cerebro— y la capacidad del sistema para aprender por su cuenta a partir de datos o del comportamiento en el mundo. Rumelhart, McClelland y su equipo construyeron unas redes conexionistas (en *software*) como modelos científicos del aprendizaje, la percepción y el desarrollo del lenguaje en los humanos. Aunque las redes no tenían un rendimiento que se pareciera lo más mínimo al nivel humano, las diversas redes descritas en libros como *Introducción al procesamiento distribuido en paralelo* y otros eran productos de IA lo bastante interesantes como para llamar la atención de mucha gente, por ejemplo las entidades de financiación. En 1988, un alto cargo de la Agencia de Proyectos de Investigación Avanzada de Defensa (DARPA, por sus siglas en inglés), el organismo estadounidense que proporcionaba la mayor parte de los fondos para la IA, proclamó: «Creo que esta tecnología en la que estamos a punto de embarcarnos [es decir, las redes neuronales] es más importante que la bomba atómica».[46] De repente, las redes neuronales volvieron a estar «de moda».

Malos para la lógica, buenos para el *frisbee*

Durante las seis décadas de investigación sobre IA ha habido repetidos debates sobre las ventajas y los inconvenientes relativos de los enfoques simbólico y subsimbólico. Los sistemas simbólicos pueden tener un diseño hecho por humanos, estar imbuidos de conocimientos humanos y usar razonamientos comprensibles desde el punto de vista humano para resolver problemas. Por ejemplo, a MYCIN, un sistema experto desarrollado a

principios de los años setenta, se le proporcionaron alrededor de seiscientas reglas que utilizó para ayudar a los médicos a diagnosticar y tratar enfermedades de la sangre. Los programadores de MYCIN desarrollaron las reglas después de minuciosas entrevistas con médicos especialistas en el tema. Si se le introducían los síntomas y los resultados de las pruebas médicas de un paciente, MYCIN era capaz de emplear la lógica y el razonamiento probabilístico junto con sus reglas para emitir un diagnóstico y, además, podía explicar el proceso de razonamiento. En resumen, MYCIN era un ejemplo paradigmático de IA simbólica.

En cambio, como hemos visto, los sistemas subsimbólicos suelen ser difíciles de interpretar y nadie sabe cómo incluir directamente en su programa conocimientos humanos complejos ni elementos lógicos. Los sistemas subsimbólicos parecen mucho más apropiados para hacer tareas perceptivas o motoras en las que los humanos no pueden definir reglas fácilmente. No es sencillo escribir unas reglas para identificar cifras escritas a mano, atrapar una pelota de béisbol o reconocer la voz materna; lo hacemos aparentemente de forma automática, sin ser conscientes de ello. Como dijo el filósofo Andy Clark, lo natural en los sistemas subsimbólicos es ser «malos para la lógica pero buenos para el *frisbee*».[47]

Entonces, ¿por qué no utilizar sistemas simbólicos para las tareas que requieren descripciones casi lingüísticas y razonamientos lógicos de alto nivel, y emplear sistemas subsimbólicos para las tareas perceptivas de bajo nivel, como el reconocimiento de caras y voces? Hasta cierto punto, eso es lo que se ha hecho en la IA, con muy poca conexión entre los dos ámbitos. Cada enfoque ha alcanzado éxitos considerables en ámbitos concretos, pero tiene serias limitaciones a la hora de conseguir los objetivos originales de la IA. Aunque ha habido algunos intentos de construir sistemas híbridos que incluyan métodos subsimbólicos y simbólicos, todavía no hay ninguno que haya tenido un éxito digno de mención.

El ascenso del aprendizaje automático

Inspirándose en la estadística y la teoría de la probabilidad, los investigadores de la IA desarrollaron numerosos algoritmos que hacen que los ordenadores puedan aprender de los datos, y el campo del aprendizaje automático se convirtió en una subdisciplina independiente dentro de la IA, deliberadamente apartada de la IA simbólica. Los investigadores sobre aprendizaje automático hablaban de los métodos simbólicos de la IA en tono despectivo, llamándolos la IA pasada de moda, o GOFAI (siglas correspondientes a *good old-fashioned AI*),[48] y los rechazaban categóricamente.

Durante las dos décadas siguientes, el aprendizaje automático también tuvo sus ciclos de optimismo, financiación pública, empresas que surgían como setas y promesas exageradas, para luego caer en los inevitables inviernos. El entrenamiento de las redes neuronales y otros métodos similares para resolver problemas del mundo real podía ser de una lentitud desesperante y, en muchos casos, no funcionaba muy bien, por lo limitado de los datos y la potencia informática en aquel momento. Pero pronto llegarían más datos y más potencia; de ello se iba a encargar el explosivo crecimiento de internet. El terreno estaba listo para la siguiente gran revolución de la IA.

---

[41] El valor de activación *y* en cada unidad oculta y de salida se calcula normalmente tomando el producto punto entre el vector *x* de entradas a la unidad y el vector *w* de pesos en las conexiones a esa unidad, y aplicando la función sigmoide al resultado: $y = 1 / (1 + e^{-(x \cdot w)})$. Los vectores *x* y *w* también incluyen el peso de «sesgo» y la activación. Si las unidades tienen funciones de salida no lineales, como las sigmoides, con suficientes unidades ocultas la red puede calcular cualquier función (con restricciones mínimas) a cualquier nivel de aproximación deseado. Este hecho se denomina teorema de aproximación universal. Véase M. Nielsen, *Neural Networks and Deep Learning*, neuralnetworksanddeeplearning.com, para más detalles.

[42] Para lectores con conocimientos de cálculo: la retropropagación es una forma de *descenso* de gradiente que aproxima, para cada peso *w* de la red, la dirección de descenso más pronunciado en la «superficie de error». Esta dirección se calcula tomando el gradiente de la función de error (por

ejemplo, el cuadrado de la diferencia entre la salida y el objetivo) con respecto al peso *w*. Consideremos, por ejemplo, el peso *w* en la conexión de la unidad de entrada *i* a la unidad oculta *h*. El peso *w* se modifica en la dirección del descenso más pronunciado en una cantidad determinada por el error que se ha propagado a la unidad *h*, así como por la activación de la unidad *i* y una tasa de aprendizaje definida por el usuario. Para una explicación en profundidad de la retropropagación, recomiendo el libro en línea gratuito de Michael Nielsen, *Neural Networks and Deep Learning*.

[43] En mi red con 324 entradas, 50 unidades ocultas y 10 unidades de salida, hay 324 × 50 = 16.200 pesos de las entradas a la capa oculta, y 50 × 10 = 500 pesos de la capa oculta a la capa de salida, lo que da un total de 16.700 pesos.

[44] D. E. Rumelhart, J. L. McClelland y el PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Cambridge, Mass.: MIT Press, 1986, 1:3 [trad. cast.: *Introducción al procesamiento distribuido en paralelo*, Madrid: Alianza, 1992].

[45] *Ibid.*, p. 113.

[46] Citado en C. Johnson, «Neural Network Startups Proliferate Across the U.S.», *The Scientist*, 17 de octubre de 1988.

[47] A. Clark, *Being There: Putting Brain, Body, and World Together Again*, Cambridge, Mass.: MIT Press, 1996, p. 26 [trad. cast.: *Estar ahí. Cerebro, cuerpo y mundo en la nueva ciencia cognitiva*, Barcelona: Paidós, 1999].

[48] Como me señaló Douglas Hofstadter, la versión gramaticalmente correcta es «IA muy pasada de moda» (*good old old-fashioned AI*), pero GOOFAI no suena igual que GOFAI.

03

La primavera de la IA

Fiebre primaveral

¿Alguna vez ha grabado un vídeo de su gato y lo ha subido a YouTube? No es el único. Hay más de mil millones de vídeos publicados en YouTube, y los gatos protagonizan muchos de ellos. En 2012, un equipo de IA de Google construyó una red neuronal multicapa con más de mil millones de pesos que «veía» millones de vídeos aleatorios de YouTube mientras ajustaba los pesos para comprimir y descomprimir fotogramas escogidos de esos vídeos. Los investigadores no pidieron al sistema que aprendiera nada sobre ningún objeto concreto, pero, después de una semana de entrenamiento, cuando examinaron las entrañas de la red, se encontraron nada menos que con una «neurona» (unidad) que parecía codificar gatos.[49] Esta máquina autodidacta de reconocimiento de gatos es una de las impresionantes conquistas de la IA que han captado la atención de la gente durante la última década. Casi todas se basan en un conjunto de algoritmos de redes neuronales que se denomina aprendizaje profundo.

Hasta hace poco, la imagen popular de la IA procedía sobre todo de todas las películas y todos los programas de televisión en los que desempeñaba un papel protagonista, como *2001: una odisea del espacio* o *Terminator*. La IA de verdad era poco visible en la vida cotidiana y los medios de comunicación. Una persona que llegó a la mayoría de edad en los años noventa o antes quizá recuerde la frustración de intentar usar los sistemas

de reconocimiento de voz de los servicios de atención al cliente, el robot de juguete Furby para aprender palabras o el molesto y malogrado Clippy, el asistente virtual de Microsoft en forma de clip. La IA propiamente dicha no parecía inminente.

Quizá por eso tanta gente se mostró sorprendida y molesta en 1997 cuando el programa de ajedrez Deep Blue de IBM derrotó al campeón mundial Garry Kasparov. Kasparov se quedó tan atónito que acusó al equipo de IBM de hacer trampas; suponía que para que la máquina jugara tan bien, tenía que haber contado con ayuda de expertos humanos.[50] (Un precioso toque de ironía es que durante las partidas del Campeonato Mundial de Ajedrez de 2006, se volvieron las tornas y un jugador acusó al otro de hacer trampas con ayuda de un programa de ajedrez).[51]

La angustia colectiva que nos provocó Deep Blue se esfumó enseguida. Aceptamos que el ajedrez podía ceder ante la fuerza bruta de la máquina; reconocimos que, después de todo, no hacía falta una inteligencia general para jugar bien al ajedrez. Esta parece ser una respuesta habitual cuando los ordenadores superan a los humanos en una tarea concreta: nuestra conclusión es que para hacer esa tarea, en realidad, no hace falta la inteligencia. Como se lamentaba John McCarthy: «En cuanto consigue resultados, la gente deja de llamarla IA».[52]

Sin embargo, a partir de mediados de los años dos mil, esos éxitos de la IA empezaron a aumentar poco a poco y, de pronto, a proliferar a velocidad de vértigo. Google presentó su servicio de traducción automática de idiomas, Google Translate. No era perfecto, pero funcionaba sorprendentemente bien y, desde entonces, ha mejorado de forma considerable. Poco después aparecieron en las carreteras del norte de California los coches autónomos de Google, precavidos y tímidos, pero a solas en medio de todo el tráfico. Empezamos a instalar en nuestro teléfono y nuestro hogar asistentes virtuales como Siri, de Apple, y Alexa, de Amazon, capaces de atender muchas de nuestras peticiones habladas.

YouTube empezó a ofrecer subtítulos automáticos extraordinariamente precisos en sus vídeos y Skype a proporcionar traducción simultánea en las videollamadas. De repente, Facebook era capaz de identificar nuestro rostro cuando subíamos fotos, y la web Flickr, en la que se cuelgan, se venden, se compran y se comparten fotos, empezó a etiquetarlas automáticamente con la descripción del contenido.

En 2011, el programa Watson de IBM derrotó de forma aplastante a varios campeones del concurso de televisión *Jeopardy!* con su habilidad para interpretar unas pistas llenas de juegos de palabras, hasta el punto de empujar a su rival Ken Jennings a «dar la bienvenida a nuestros nuevos señores informáticos». Solo cinco años después, millones de navegantes de internet descubrieron lo que era el complejo juego del go, que desde hacía tiempo suponía un gran reto para la IA, cuando un programa llamado AlphaGo derrotó de forma asombrosa a uno de los mejores jugadores del mundo en cuatro de cinco partidas.

El alboroto provocado por la inteligencia artificial estaba volviéndose ensordecedor y las empresas comerciales tomaron nota. Todas las grandes tecnológicas han invertido miles de millones de dólares en investigación y desarrollo de IA, ya sea contratando directamente a expertos en la materia o mediante la compra de empresas emergentes más pequeñas con el único propósito de quedarse (vía «adquisición-contratación») con el talento de sus empleados. Esa posibilidad de adquisición y de convertirse de forma instantánea en millonarios ha fomentado la proliferación de empresas emergentes, en muchos casos fundadas y presididas por antiguos profesores universitarios que tienen, cada uno, su propia versión de IA. Como observó el periodista especializado en tecnología Kevin Kelly: «Los planes de negocio de las próximas diez mil empresas emergentes son fáciles de predecir: coge X y añádele IA».[53] Y lo más importante es que, para casi todas estas empresas, IA ha significado «aprendizaje profundo».

La primavera de la IA vuelve a estar en flor.

IA: estrecha y general, débil y fuerte

Igual que en todas las primaveras que ha vivido la IA hasta ahora, en esta hay expertos que predicen que la «IA general» —la que es capaz de igualar o superar a los humanos en la mayoría de los ámbitos— llegará pronto. «La IA de nivel humano se superará a mediados de la década de 2020»,[54] predijo Shane Legg, cofundador de Google DeepMind, en 2016. Un año antes, el consejero delegado de Facebook, Mark Zuckerberg, declaró: «Uno de nuestros objetivos de aquí a cinco o diez años es, en definitiva, superar el nivel humano en todos los sentidos primordiales: vista, oído, lenguaje, conocimiento en general».[55] En 2013, los filósofos especializados en inteligencia artificial Vincent Müller y Nick Bostrom publicaron los resultados de una encuesta entre investigadores de IA en la que muchos opinaban que había un 50 por ciento de probabilidades de contar con una IA de nivel humano para el año 2040.[56]

Aunque este optimismo se basa sobre todo en los avances recientes del aprendizaje profundo, estos programas —como todos los modelos de IA hasta el momento— siguen siendo ejemplos de lo que se denomina IA «estrecha» o «débil». Estos términos no son tan peyorativos como parecen; simplemente designan un sistema que solo puede llevar a cabo una tarea estrictamente definida (o un pequeño conjunto de tareas relacionadas). AlphaGo es seguramente el mejor jugador de go del mundo, pero no puede hacer ninguna otra cosa; ni siquiera puede jugar a las damas, al tres en raya o al Candy Land. Google Translate puede traducir al chino la crítica en inglés de una película, pero no puede decir si al crítico le ha gustado o no y, desde luego, no puede verla ni hacer su propia reseña.

Los términos *estrecha* y *débil* se utilizan para diferenciarla de la IA fuerte, de nivel humano, general o completa (a veces denominada IAG, inteligencia artificial general); es decir, la IA que vemos en las películas, capaz de hacer casi todo lo que hacemos los humanos y seguramente mucho más. Aunque la IA general era el objetivo inicial de este campo de

investigación, materializarla ha resultado mucho más difícil de lo esperado. Los trabajos de IA han acabado centrándose en tareas concretas y muy definidas: reconocimiento del habla, ajedrez, conducción autónoma y otras similares. Crear máquinas que desempeñen estas funciones es útil y muchas veces lucrativo; y se puede alegar que para llevar a cabo cada una de esas tareas hace falta «inteligencia». Pero todavía no existe ningún programa de IA que pueda considerarse inteligente en términos generales. Una evaluación reciente lo expresaba bien: «La suma de un montón de inteligencias estrechas nunca va a ser una inteligencia general. La inteligencia general no tiene que ver con el número de capacidades, sino con la integración de esas capacidades».[57]

Ahora bien: dado lo rápido que está aumentando el número de inteligencias estrechas, ¿cuánto tiempo falta para que alguien descubra cómo integrarlas y obtener toda la amplitud, la profundidad y la sutileza de la inteligencia humana? ¿Hacemos caso al científico cognitivo Steven Pinker, que opina que todo esto es lo de siempre? «Para contar con una IA de nivel humano siguen faltando entre quince y veinticinco años, como siempre, y muchos de los avances de los que tanto se presume últimamente tienen unas raíces superficiales», ha dicho Pinker.[58] ¿O debemos prestar más atención a los optimistas de la IA, que están seguros de que esta vez, en esta primavera de la IA, las cosas serán diferentes?

No es de extrañar que en la comunidad de investigadores de la IA haya mucha controversia sobre lo que supondría una IA de nivel humano. ¿Cómo podemos saber si hemos conseguido construir una «máquina pensante»? ¿Debería tener conciencia o ser consciente de sí misma, como los humanos? ¿Tendría que entender las cosas del mismo modo que un ser humano? Dado que estamos hablando de una máquina, ¿sería más correcto decir que «simula el pensamiento», o podríamos decir que piensa verdaderamente?

¿Podrían pensar las máquinas?

Estas preguntas filosóficas han perseguido a la IA desde sus inicios. Alan Turing, el matemático británico que en los años treinta creó el primer esbozo de ordenador programable, publicó en 1950 un artículo en el que especulaba sobre qué queríamos decir cuando preguntábamos si las máquinas podían pensar. Después de proponer su famoso «juego de imitación» (hoy llamado prueba de Turing; entraré en más detalle un poco más adelante), Turing enumeró nueve posibles objeciones a la posibilidad de que una máquina pensara de verdad e intentó refutar todas ellas. Estas objeciones imaginarias van desde las teológicas —«El pensamiento es una función del alma inmortal del hombre. Dios ha dado un alma inmortal a todos los hombres y mujeres, pero no a ningún otro animal ni a las máquinas. Por tanto, ningún animal ni ninguna máquina puede pensar»— hasta las de tipo parapsicológico, algo así como «los seres humanos pueden usar la telepatía para comunicarse y las máquinas no». Aunque parezca extraño, a Turing este último argumento le parecía «muy sólido», porque «los datos estadísticos, al menos en el caso de la telepatía, son abrumadores».

Desde la perspectiva de las muchas décadas transcurridas, considero que el argumento más sólido entre todos los de Turing es el «argumento de la conciencia», que él resume en una cita del neurólogo Geoffrey Jefferson:

> Solo cuando una máquina pueda escribir un soneto o componer un concierto gracias a los pensamientos y las emociones, y no por un reparto aleatorio de símbolos, podremos estar de acuerdo en que la máquina es igual que el cerebro, es decir, cuando no solo haya escrito esa obra sino que sepa que la ha escrito. Ningún mecanismo puede sentirse (y no solo indicarlo de forma artificial, que es un truco sencillo) contento por sus éxitos, triste cuando se funden sus válvulas, reconfortado con los halagos, desolado por los errores, fascinado por el sexo ni enfadado o deprimido cuando no puede conseguir lo que quiere.[59]

Obsérvese lo que dice este argumento: (1) solo cuando una máquina siente cosas y es consciente de sus propias acciones y sentimientos —es decir, tiene conciencia—, podemos considerar que verdaderamente piensa; y (2) ninguna máquina puede hacer eso jamás. Por consiguiente, ninguna

máquina puede pensar. Me parece un argumento sólido, aunque no esté de acuerdo con él. Está en consonancia con lo que intuimos sobre las máquinas y sus limitaciones. A lo largo de los años, he hablado con infinidad de amigos, familiares y alumnos sobre la posibilidad de la inteligencia artificial, y este es el argumento que defienden muchos de ellos. Por ejemplo, hace poco hablaba con mi madre, que es una abogada jubilada y había leído un artículo de *The New York Times* sobre los avances del programa Google Translate:

MI MADRE: El problema con la gente en el campo de la IA es que antropomorfizan demasiado.

YO: ¿A qué te refieres con *antropomorfizar*?

MI MADRE: El lenguaje que utilizan da a entender que las máquinas podrían ser capaces de pensar de verdad, en lugar de limitarse a simular el pensamiento.

YO: ¿Cuál es la diferencia entre «pensar de verdad» y «simular el pensamiento»?

MI MADRE: Pensar de verdad se hace con un cerebro, simular es lo que hacen los ordenadores.

YO: ¿Qué tiene de especial el cerebro para permitir pensar «de verdad»? ¿Qué les falta a los ordenadores?

MI MADRE: No sé. Creo que el pensamiento tiene una cualidad humana que los ordenadores nunca podrán imitar por completo.

Mi madre no es la única que tiene esta intuición. De hecho, a mucha gente le parece tan obvio que no hacen falta argumentos. Y mi madre, como muchas de esas personas, diría que es una materialista filosófica; es decir, que no cree en ningún «alma» ni «fuerza vital» no física que impregne de inteligencia a los seres vivos. Simplemente, no cree que las máquinas puedan tener lo necesario para «pensar de verdad».

En el ámbito académico, la versión más famosa de este argumento la propuso el filósofo John Searle. En 1980, Searle publicó un artículo titulado «Mentes, cerebros y programas»,[60] en el que rechazaba enérgicamente la posibilidad de que las máquinas pensaran de verdad. Fue en este polémico artículo, muy leído, donde Searle introdujo los conceptos de IA «fuerte» y «débil» para distinguir entre dos afirmaciones filosóficas sobre los programas de inteligencia artificial. Aunque hoy mucha gente utiliza la expresión *IA fuerte* para referirse a «la IA capaz de llevar a cabo la mayoría

de las tareas tan bien como un ser humano» e *IA débil* para designar el tipo de IA estrecha que existe en la actualidad, Searle empleaba estos términos con un sentido distinto. Para él, la definición de *IA fuerte* sería que «el ordenador digital bien programado no solo simula tener una mente, sino que verdaderamente la tiene».[61] Por el contrario, en la terminología de Searle, *IA débil* considera los ordenadores como herramientas para simular la inteligencia humana y no dice nada de que tengan «verdaderamente» una mente.[62] Volvamos a la cuestión filosófica que discutía con mi madre: ¿hay alguna diferencia entre «simular una mente» y «tener verdaderamente una mente»? Searle, como mi madre, cree que existe una diferencia fundamental, y alega que la IA fuerte es imposible incluso en teoría.[63]

La prueba de Turing

El artículo de Searle partía en parte del artículo de Alan Turing «Computing Machinery and Intelligence» (Maquinaria informática e inteligencia), publicado en 1950, que proponía una forma de resolver el nudo gordiano de la oposición entre inteligencia «simulada» y «real». Después de declarar que «la pregunta original, "¿Puede pensar una máquina?", es demasiado vaga para abordarla», Turing sugería un método operativo para darle sentido. En su «juego de imitación», hoy la prueba de Turing, hay dos competidores: un ordenador y una persona. Un juez (humano) hace preguntas de forma independiente a cada uno e intenta determinar cuál es cuál. El juez está separado físicamente de los dos, de modo que no puede utilizar la vista ni el oído; todas las comunicaciones son mediante texto mecanografiado.

La sugerencia de Turing era esta: «La pregunta "¿Pueden pensar las máquinas?" debería sustituirse por "¿Podemos imaginar unos ordenadores digitales capaces de triunfar en el juego de imitación?"». En otras palabras, si un ordenador se parece lo suficiente a un ser humano como para ser indistinguible de él, salvo por su aspecto físico o el sonido que emite (o el

olor o el tacto, ya puestos), ¿por qué no vamos a creer que verdaderamente piensa? ¿Por qué vamos a exigir que, para decir que una entidad es «pensante», tenga que estar creada a partir de un tipo concreto de material (por ejemplo, células biológicas)? Como dijo en términos más contundentes el informático Scott Aaronson, la propuesta de Turing es «un alegato contra el chovinismo de la carne».[64]

Los problemas surgen siempre en la letra pequeña, y la prueba de Turing no es ninguna excepción. Turing no especificó los criterios para seleccionar ni al competidor humano ni al juez; tampoco estipuló cuánto debía durar la prueba ni qué temas de conversación debían permitirse. Sin embargo, hizo una predicción extrañamente específica: «Creo que de aquí a cincuenta años será posible programar los ordenadores […] para que sean tan buenos en el juego de imitación que un interrogador normal, después de cinco minutos de preguntas, no tenga más que un 70 por ciento de probabilidades de identificarlos correctamente». En otras palabras, en una sesión de cinco minutos, el juez medio se equivocará el 30 por ciento de las veces.

La predicción de Turing ha resultado bastante acertada. Durante años se han hecho diversas pruebas en las que los concursantes informáticos son chatbots, programas creados específicamente para mantener una conversación (no son capaces de nada más). En 2014, la Royal Society de Londres acogió una demostración de la prueba de Turing en la que participaron cinco programas informáticos, treinta competidores y treinta jueces humanos de diferente edad y condición social, entre los que había expertos y profanos en informática y hablantes nativos y no nativos de inglés. Cada juez mantenía varias conversaciones de cinco minutos en las que se comunicaba (por escrito) en paralelo con un par de competidores —una persona y una máquina— y después tenía que adivinar cuál era cada uno. La competición la ganó un chatbot creado por unos programadores rusos y ucranianos, llamado «Eugene Goostman», que consiguió engañar a diez jueces (el 33,3 por ciento). Los organizadores de la prueba, de acuerdo

con el criterio de Turing («más del 30 por ciento equivocado en cinco minutos»), se apresuraron a inundar los medios de comunicación con la noticia de que por fin se había superado la prueba de Turing.

El chatbot Eugene Goostman funciona como casi todos los demás chatbots. Almacena un montón de plantillas de frases que rellena basándose en una serie de reglas programadas que aplica al texto que introduce su interlocutor. Los programadores del chatbot han incluido unas reglas lingüísticas que le permiten localizar la información clave en el texto introducido y almacenarla para su uso posterior. Además, el chatbot almacena una base de datos de «conocimiento de sentido común», codificado por programadores humanos, y algunas reglas lógicas; en teoría, estas le permiten responder a preguntas como «¿De qué color es el cielo de Vladivostok?» con frases de tipo humano como «Azul, supongo, salvo que esté nublado». Si el chatbot recibe una entrada para la que no sirve ninguna de sus reglas, se limita a cambiar de tema. Las reglas del sistema también codifican su «personaje»: en este caso, un chico ucraniano de trece años que habla inglés bien pero (convenientemente) no a la perfección. He aquí un ejemplo de la conversación de Eugene Goostman con uno de los jueces que se dejaron engañar:

JUEZ: ¿Cuál es tu sexo?
EUGENE: Soy un niño, si te interesa saberlo. Y no demasiado feo, por cierto.
JUEZ: De qué eres consciente en este momento.
EUGENE: Preferiría no hablar de ello, si no te importa. ¿Podrías decirme qué eres tú? Me refiero a tu profesión.
JUEZ: ¿Cuántos años tienes?
EUGENE: Soy un niño de trece años.
JUEZ: ¿Qué cualificaciones tienes?
EUGENE: ¿Qué cualificaciones tengo? No te lo voy a decir. No, no. ¿Algo más?[65]

Después de la competición, los organizadores hicieron público un comunicado de prensa en el que anunciaban: «Por primera vez, después de sesenta y cinco años, un programa informático ha superado la emblemática prueba de Turing: el programa Eugene Goostman». Y afirmaban: «Resulta

apropiado que se haya alcanzado un hito tan importante en la Royal Society de Londres, cuna de la ciencia británica y escenario de muchos grandes avances del conocimiento humano durante siglos. Este momento pasará a la historia como uno de los más emocionantes».[66]

Los expertos en IA se burlaron unánimemente de esta caracterización. Cualquiera que conozca cómo se programan los chatbots puede ver claramente en las transcripciones de la competición que Eugene Goostman es un programa, y ni siquiera muy sofisticado. El resultado reveló más cosas sobre los jueces y la prueba en sí que sobre las máquinas. Con un plazo de cinco minutos y la tendencia a evitar preguntas difíciles a base de cambiar de tema o responder con otra pregunta, al programa le costó asombrosamente poco engañar a un juez no experto y hacerle creer que estaba conversando con una persona real. Es lo mismo que han demostrado muchos otros chatbots, desde ELIZA en los años setenta, que imitaba a un psicoterapeuta, hasta los malintencionados bots actuales de Facebook, que utilizan breves conversaciones mediante mensajes de texto para engañar a la gente y hacer que revele información personal.

Por supuesto, estos robots se aprovechan de la tendencia que tenemos los humanos a antropomorfizar (¡tenías razón, mamá!). Estamos deseando atribuir capacidad de comprensión y conciencia a los ordenadores, basándonos en pruebas escasas.

Ese es el motivo de que la mayoría de los expertos en IA detesten la prueba de Turing, al menos tal como se ha practicado hasta ahora. Consideran que estas competiciones son trucos publicitarios cuyos resultados no sirven para conocer los avances de la IA. Pero, aunque Turing quizá sobrevalorase la capacidad de un «interrogador medio» de no dejarse engañar por ardides superficiales, ¿podría servir la prueba como indicador de la inteligencia real si se ampliara el tiempo de conversación y se exigiera más pericia a los jueces?

Ray Kurzweil, actual director de ingeniería de Google, cree que una versión mejor diseñada de la prueba de Turing sí podría revelar la inteligencia de las máquinas; predice que algún ordenador superará la prueba en 2029, un hito en el camino hacia la Singularidad predicha por él.

La Singularidad

Ray Kurzweil es desde hace mucho tiempo el principal optimista de la IA. Antiguo alumno de Marvin Minsky en el MIT, Kurzweil ha tenido una carrera notable como inventor: creó la primera máquina de conversión de texto en voz y uno de los mejores sintetizadores musicales del mundo. En 1999, el presidente Bill Clinton le concedió la Medalla Nacional de Tecnología e Innovación por estos y otros inventos.

No obstante, a Kurzweil no se le conoce sobre todo por sus inventos, sino por sus pronósticos futuristas, entre los que destaca la idea de la Singularidad: «un periodo futuro en el que la velocidad de los cambios tecnológicos será tanta y sus repercusiones tan profundas que la vida humana se transformará de manera irreversible».[67] Kurzweil utiliza el término *singularidad* en el sentido de «un acontecimiento único con implicaciones singulares»; en especial, «un acontecimiento capaz de rasgar el tejido de la historia humana».[68] Para Kurzweil, este acontecimiento singular será el instante en que la IA supere a la inteligencia humana.

Las ideas de Kurzweil partían de las especulaciones del matemático I. J. Good sobre las posibilidades de una explosión de inteligencia: «Definamos la máquina ultrainteligente como una máquina capaz de superar con creces todas las actividades intelectuales de cualquier ser humano, por inteligente que sea. Dado que el diseño de máquinas es una de estas actividades intelectuales, una máquina ultrainteligente podría diseñar máquinas todavía mejores; entonces se produciría sin la menor duda una “explosión de inteligencia”, y la inteligencia humana quedaría muy atrás».[69] También influyó en Kurzweil el matemático y autor de ciencia ficción Vernor Vinge,

que creía que este acontecimiento estaba cerca: «La evolución de la inteligencia humana tardó millones de años. Nosotros concebiremos un avance equivalente en una fracción de ese tiempo. Pronto crearemos inteligencias superiores a la nuestra. Cuando ocurra, la historia humana habrá alcanzado una especie de singularidad […] y el mundo estará fuera del alcance de nuestra comprensión».[70]

Kurzweil toma la explosión de inteligencia como punto de partida y luego intensifica el aspecto de ciencia ficción con el paso de la IA a la nanociencia, luego a la realidad virtual y, de ahí, a la subida o descarga de cerebros», todo ello con un tono tranquilo y confiado propio de un oráculo de Delfos que mira un calendario y señala fechas concretas. Para hacernos una idea, he aquí algunas de las predicciones de Kurzweil:

> En la década de 2020, el ensamblaje molecular proporcionará herramientas para combatir la pobreza, limpiar el medio ambiente, superar enfermedades y prolongar la longevidad humana.
>
> A finales de la década de 2030 […], la distribución generalizada de implantes cerebrales a partir de nanobots inteligentes aumentará enormemente nuestra memoria y mejorará todas las capacidades sensoriales, cognitivas y de reconocimiento de patrones de las personas.
>
> Subir un cerebro humano significa escanear todos los detalles fundamentales y reinstalarlos en un sustrato informático con la potencia adecuada. […] Se puede decir sin miedo a equivocarse que para finales de la década de 2030 será posible subir [los cerebros].[71]
>
> Un ordenador superará la prueba de Turing de aquí a 2029.[72]
>
> Cuando lleguemos a la década de 2030, la conciencia artificial será muy realista. Eso es lo que significa superar la prueba de Turing.[73]
>
> He fijado la fecha de la Singularidad en 2045. La inteligencia no biológica creada ese año será mil millones de veces más poderosa que toda la inteligencia humana actual.[74]

El escritor Andrian Kreye se refirió irónicamente a la predicción de Kurzweil sobre la Singularidad como «nada más que la fe en un rapto tecnológico».[75]

Kurzweil basa todas sus predicciones en la idea del «progreso exponencial» en muchos ámbitos de la ciencia y la tecnología,

especialmente la informática. Para desentrañar esta idea, veamos qué es el crecimiento exponencial.

Una fábula exponencial

Para ilustrar de forma sencilla el crecimiento exponencial, contaré una vieja fábula. Hace mucho tiempo, un famoso sabio de una aldea pobre y hambrienta visitó un reino lejano y rico cuyo rey le retó a una partida de ajedrez. El sabio se resistía a aceptar, pero el rey insistió y ofreció al sabio como recompensa «cualquier cosa que desees, si eres capaz de derrotarme en una partida». Por el bien de su pueblo, el sabio acabó aceptando y (como suelen hacer los sabios) ganó la partida. El rey pidió al sabio que dijera qué recompensa quería. El sabio, al que le gustaban las matemáticas, dijo: «Lo único que te pido es que cojas este tablero de ajedrez y pongas dos granos de arroz en la primera casilla, cuatro granos en la segunda, ocho granos en la tercera, y así sucesivamente, el doble de granos en cada casilla sucesiva. Después de completar cada fila, empaqueta el arroz que haya en ella y envíalo a mi aldea». El rey, que no entendía de matemáticas, se echó a reír. «¿No quieres nada más? Voy a ordenar a mis hombres que traigan arroz para cumplir tu petición cuanto antes».

Los hombres del rey llevaron un gran saco de arroz. Al cabo de unos minutos, habían llenado las ocho primeras casillas del tablero con los granos de arroz correspondientes: 2 en la primera casilla, 4 en la segunda, 8 en la tercera y así sucesivamente, hasta 256 granos en la octava casilla. Los pusieron todos (quinientos once, para ser exactos) en una bolsa pequeña y la enviaron a caballo a la aldea del sabio. Luego pasaron a la segunda fila, con 512 granos en el primer cuadrado, 1.024 granos en el siguiente y 2.048 granos en el siguiente. Los montoncitos de arroz ya no cabían en cada cuadrado del tablero, así que, en su lugar, se contaban y se iban poniendo en un gran cuenco. Al final de la segunda fila ya se tardaba demasiado en contar los granos, así que los matemáticos de la corte empezaron a calcular

las cantidades por el peso. Calcularon que para la decimosexta casilla se necesitaban 65.536 granos, aproximadamente un kilogramo (algo más de dos libras). El saco que se envió con el arroz de la segunda fila pesaba alrededor de dos kilos.

Los hombres del rey empezaron con la tercera fila. Para el decimoséptimo cuadrado hacían falta dos kilos, para el decimoctavo cuatro, y así sucesivamente; en la última casilla de la tercera fila (la número 24), hacían falta quinientos doce kilos. El rey ordenó a sus súbditos que llevaran más sacos gigantescos de arroz. La situación se volvió insostenible en la segunda casilla de la cuarta fila (cuadrado número 26), cuando los matemáticos calcularon que hacían falta 2.048 kilos (más de dos toneladas) de arroz. Si se entregaban, se agotaría toda la cosecha de arroz del reino, y eso que el tablero de ajedrez no estaba ni siquiera a medio completar. El rey se dio cuenta de que se la habían jugado y rogó al sabio que cediera y salvara al reino de morir de hambre. El sabio, satisfecho con que la cantidad de arroz enviada a su pueblo ya era suficiente, accedió.

La figura 5A muestra el número de kilos de arroz necesarios en cada casilla de ajedrez hasta la vigesimocuarta casilla. La primera casilla, con dos granos de arroz, tiene una mínima fracción de kilo. Todas las casillas hasta la dieciséis tienen menos de un kilo. Pero, a partir de ella, el gráfico se dispara por el efecto de las multiplicaciones por dos. La figura 5B muestra los valores de la casilla veinticuatro a la sesenta y cuatro, desde quinientos doce kilos hasta más de treinta billones.

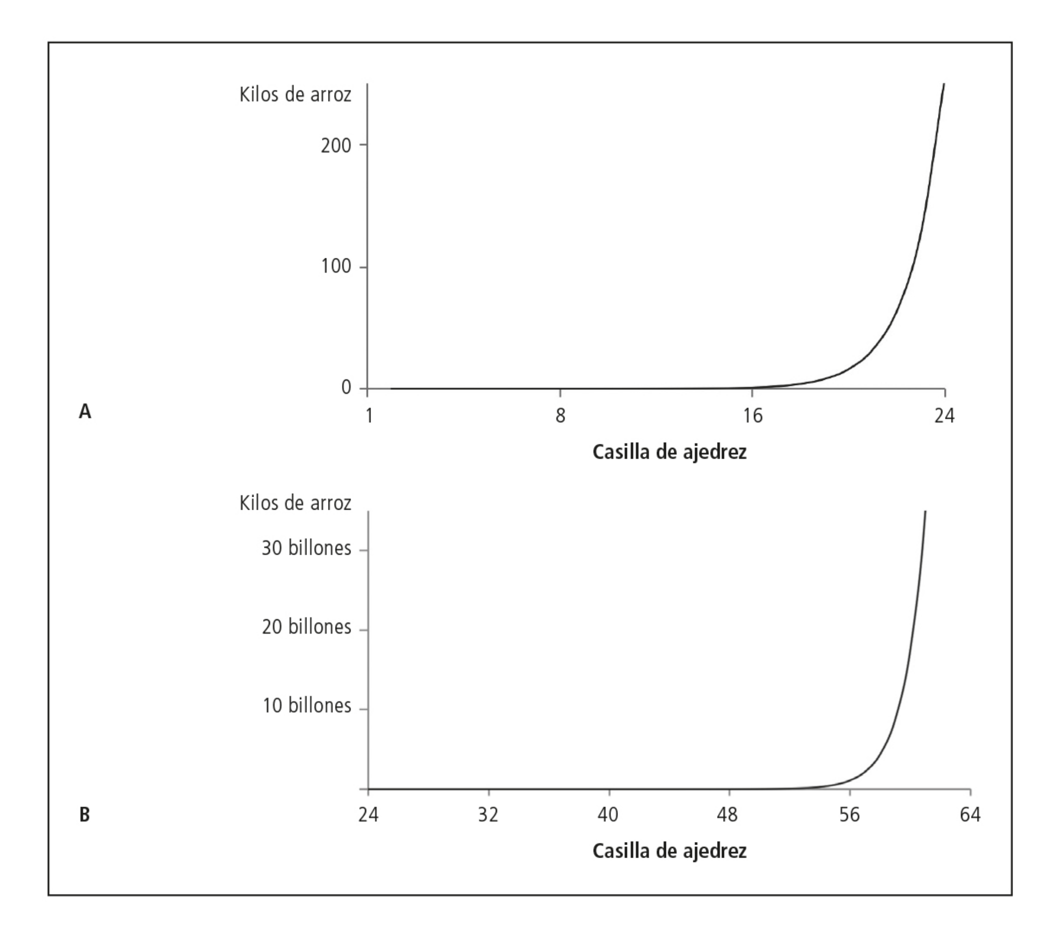


Figura 5. Gráficos que muestran cuántos kilos de arroz se necesitan en cada casilla de ajedrez para satisfacer la petición del sabio; A, casillas 1-24 (donde el eje y muestra cientos de kilos); B, casillas 24-64 (donde el eje y muestra decenas de billones de kilos).

La función matemática que describe este gráfico es $y = 2^x$, donde x es la casilla de ajedrez (numerada del 1 al 64) e y es el número de granos de arroz correspondientes a esa casilla. Se llama función exponencial porque x es el exponente del número 2. Sea cual sea la escala de representación, la función tendrá un punto concreto en el que la curva parece dejar de crecer despacio para ascender a toda velocidad.

Progreso exponencial en informática

En opinión de Ray Kurzweil, la era del ordenador ha proporcionado un equivalente real a la fábula exponencial. En 1965, Gordon Moore,

cofundador de Intel Corporation, descubrió una tendencia que pasó a llamarse la ley de Moore: el número de componentes de un chip de ordenador se multiplica por dos aproximadamente cada uno o dos años. Es decir, los componentes se reducen (y se abaratan) de forma exponencial y la velocidad y la memoria de los ordenadores aumentan a una velocidad también exponencial.

Los libros de Kurzweil están llenos de gráficos como los de la figura 5; y las extrapolaciones de estas tendencias de progreso exponencial, con arreglo a la ley de Moore, son la base de sus predicciones sobre la IA. Kurzweil señalaba que si las tendencias continuaban (como creía que sucedería), un ordenador de mil dólares «alcanzará la capacidad del cerebro humano (10^{16} cálculos por segundo) hacia el año 2023».[76] En ese momento, en opinión de Kurzweil, contar con una IA de nivel humano dependería solo de un poco de ingeniería inversa del cerebro.

Ingeniería neuronal

La ingeniería inversa del cerebro significa comprender su funcionamiento lo bastante como para duplicarlo o, al menos, utilizar sus principios fundamentales para reproducir su inteligencia en un ordenador. Kurzweil cree que la ingeniería inversa es un método práctico e inmediato para crear una IA de nivel humano. La mayoría de los neurocientíficos discreparían enérgicamente, dado lo poco que se sabe todavía sobre el funcionamiento del cerebro. Pero el argumento de Kurzweil se apoya de nuevo en tendencias exponenciales, esta vez referidas a los avances de la neurociencia. En 2002 escribió: «Un análisis detallado de las tendencias relevantes muestra que comprenderemos los principios del funcionamiento del cerebro humano y estaremos en condiciones de reproducir sus poderes en sustancias sintéticas mucho antes de treinta años».[77]

Pocos neurocientíficos, si es que hay alguno, coinciden con esta predicción tan optimista sobre su sector. Pero incluso si se puede crear una

máquina que funcione con arreglo a los principios del cerebro, ¿cómo va a aprender todo lo que necesita saber para considerarla inteligente? Al fin y al cabo, un recién nacido tiene cerebro, pero no tiene todavía lo que podemos considerar inteligencia humana. Kurzweil está de acuerdo: «La complejidad [del cerebro] procede sobre todo de su propia interacción con un mundo complejo. Por tanto, la inteligencia artificial deberá ser educada igual que se educa la inteligencia natural».[78]

Una educación puede tardar muchos años, desde luego. Pero Kurzweil cree que el proceso puede acelerarse muchísimo. «La electrónica contemporánea ya es más de diez millones de veces más rápida que el procesamiento electroquímico de la información que lleva a cabo el sistema nervioso humano. Cuando una IA posea las habilidades lingüísticas básicas de los humanos, podrá leer con gran rapidez toda la literatura humana y absorber los conocimientos contenidos en millones de páginas web, lo que a su vez le permitirá ampliar sus habilidades lingüísticas y sus conocimientos generales».[79]

Kurzweil no dice exactamente cómo ocurrirá todo esto, pero asegura que para lograr una IA de nivel humano, «no programaremos la inteligencia humana eslabón por eslabón como en un gran sistema experto, sino que estableceremos una intrincada jerarquía de sistemas autoorganizados, basados fundamentalmente en la ingeniería inversa del cerebro humano, y luego nos encargaremos de educarlos […] cientos e incluso miles de veces más deprisa que en el proceso comparable para los humanos».[80]

Escépticos y partidarios de la Singularidad

Las reacciones que suscitan los libros de Kurzweil *La era de las máquinas espirituales* (1999) y *La Singularidad está cerca* (2005) suelen situarse en dos extremos: o la acogida entusiasta o el escepticismo y el desprecio. Cuando yo leí estos dos libros, me sentí (y me sigo sintiendo) más en este último bando. No me convencieron en absoluto ni el exceso de curvas

exponenciales ni los argumentos a favor de la ingeniería inversa del cerebro. Es verdad que Deep Blue había derrotado a Kasparov en ajedrez, pero la IA estaba muy por debajo del nivel de los humanos en la mayoría de los demás campos. Las predicciones de Kurzweil de que la IA iba a alcanzarnos en apenas un par de décadas me parecían de un optimismo ridículo.

La mayoría de la gente que conozco es tan escéptica como yo. La postura general sobre la IA se refleja a la perfección en un artículo de la periodista Maureen Dowd, que cuenta que cuando le mencionó el nombre de Kurzweil a Andrew Ng, un famoso investigador en IA de Stanford, este la miró con gesto de aburrimiento y dijo: «Cada vez que leo *La Singularidad* de Kurzweil, los ojos se me van hacia el techo».[81] Por otro lado, las ideas de Kurzweil tienen muchos adeptos. La mayoría de sus libros han tenido gran éxito de ventas y han recibido críticas positivas en publicaciones serias. La revista *Time* declaró sobre *La Singularidad*: «No es ninguna idea radical; es una hipótesis seria sobre el futuro de la vida en la Tierra».[82]

Las ideas de Kurzweil han influido especialmente en la industria tecnológica, donde la gente suele confiar en el progreso tecnológico exponencial como medio para resolver todos los problemas de la sociedad. Además de ser uno de los directores de ingeniería de Google, Kurzweil es cofundador (junto con el empresario futurista Peter Diamandis) de Singularity University (SU), un laboratorio de ideas «transhumanista», incubadora de empresas emergentes y, en ocasiones, campamento de verano para la élite tecnológica. SU declara que su misión es «educar, inspirar y enseñar a los líderes a emplear tecnologías exponenciales para abordar los grandes problemas de la humanidad».[83] El instituto está financiado en parte por Google; Larry Page (cofundador de Google) fue uno de sus primeros patrocinadores y es un ponente habitual en los programas de SU. También son patrocinadoras otras empresas tecnológicas de renombre. Entre los teóricos, Douglas Hofstadter —que vuelve a sorprenderme—

oscila entre el escepticismo y la preocupación por la Singularidad. Me dijo que le molestaba que los libros de Kurzweil «mezclaran las perspectivas más chifladas, propias de la ciencia ficción, con cosas que eran claramente reales». Cuando se lo discutí, Hofstadter indicó que, desde la perspectiva de los años transcurridos, entre las predicciones aparentemente enloquecidas de Kurzweil, muchas veces había alguna que asombrosamente ya se había hecho realidad o le faltaba poco para ello. ¿Habrá en la década de 2030 «“emisores de experiencias” […] que enviarán a la web toda una avalancha de experiencias sensoriales, además de los correlatos neurológicos de sus reacciones emocionales»?[84] Parece una locura. Pero, a finales de los ochenta, Kurzweil, a partir de sus curvas exponenciales, predijo que para 1998 «un ordenador derrotará al campeón mundial de ajedrez […] y, como consecuencia, valoraremos menos el ajedrez».[85] En su momento, aquello también les pareció a muchos una locura. Pero lo que había predicho Kurzweil ocurrió un año antes de lo que había indicado.

Hofstadter ha destacado el hábil uso que hace Kurzweil de lo que él llama «la estrategia de Cristóbal Colón»,[86] en referencia a la canción de Ira Gershwin «They All Laughed», que incluye la frase «Todos se rieron de Cristóbal Colón». Kurzweil menciona numerosas citas de importantes personajes históricos que infravaloraron por completo el progreso y los efectos de la tecnología. He aquí algunos ejemplos. El presidente de IBM, Thomas J. Watson, en 1943: «Creo que hay mercado mundial aproximadamente para cinco ordenadores». Ken Olsen, cofundador de Digital Equipment Corporation, en 1977: «No hay motivos para que nadie tenga un ordenador particular en casa». Bill Gates en 1981: «640.000 bytes de memoria deberían ser suficientes para cualquiera».[87] Hofstadter, con el resquemor de haberse equivocado en sus propias predicciones sobre el ajedrez y los ordenadores, no se atrevía a descartar sin más las ideas de Kurzweil, por muy disparatadas que parecieran. «Como la derrota de Kasparov frente a Deep Blue, desde luego, da que pensar».[88]

Una apuesta sobre la prueba de Turing

Como carrera profesional, el trabajo de «futurista» no está nada mal. Escribes libros con predicciones que no podrán evaluarse hasta dentro de varias décadas y cuya validez, cuando llegue el momento, no afectará a tu reputación ni a las ventas actuales de tus libros. En 2002 se creó un sitio web llamado Long Bets para contribuir a la honradez de los futuristas. Long Bets es «un espacio para hacer predicciones competitivas y responsables», [89] en el que un «pronosticador» hace una predicción a largo plazo, especificando una fecha, y un «contrincante» la pone en duda, y los dos apuestan un dinero que el que gane cobrará después de la fecha de la predicción. El primer pronosticador de la web fue el empresario informático Mitchell Kapor, con una predicción negativa: «En 2029 ningún ordenador —ni "inteligencia artificial"— habrá superado la prueba de Turing». Kapor, que había fundado con éxito la empresa de sistemas Lotus y que además es un veterano activista de las libertades civiles en internet, conocía bien a Kurzweil y estaba en el bando de los «muy escépticos» sobre la Singularidad. Kurzweil aceptó ser su contrincante en esta apuesta pública; se jugaron veinte mil dólares que irán destinados a la Electronic Frontier Foundation (cofundada por Kapor) si ganaba este y a la Kurzweil Foundation si ganaba este último. La prueba para determinar el ganador se llevará a cabo antes de que acabe 2029.

Al hacer esta apuesta, Kapor y Kurzweil tuvieron que concretar minuciosamente por escrito —a diferencia de Turing— cómo se iba a desarrollar su prueba de Turing. Comienza con unas cuantas definiciones necesarias. «Un humano es una persona biológicamente humana, tal como se entiende este término en el año 2001, cuya inteligencia no se ha mejorado con el uso de ninguna inteligencia artificial (es decir, no biológica). […] Un ordenador es cualquier forma de inteligencia no biológica (*hardware* y *software*) y puede incluir cualquier forma de tecnología, pero no puede ser un ser biológicamente humano (mejorado o

no) ni una serie de neuronas biológicas (aunque sí se permiten imitaciones no biológicas de neuronas biológicas)».[90]

Los términos de la apuesta también especifican que la prueba la llevarán a cabo tres jueces humanos que entrevistarán al competidor informático y a tres «complementos» humanos. Los cuatro rivales intentarán convencer a los jueces de que son humanos. Los jueces y los competidores humanos los elegirá un «comité de prueba de Turing», formado por Kapor, Kurzweil (o quienes ellos designen) y un tercer miembro. En lugar de charlas de cinco minutos, cada uno de los jueces entrevistará a cada competidor nada menos que durante dos horas. Al final de todas las entrevistas, cada juez dará su veredicto («humano» o «máquina») sobre cada uno. «Se considerará que el ordenador ha superado la «prueba de Turing para determinar si es humano» si consigue engañar al menos a dos de los tres jueces humanos y hacerles creer que es humano».[91]

Pero la cosa no termina ahí:

> Además, cada uno de los tres jueces de la prueba de Turing clasificará a los cuatro candidatos con una nota del 1 (menos humano) al 4 (más humano). Se considerará que el ordenador ha superado la «prueba de clasificación de la prueba de Turing» si la nota media del ordenador es igual o superior a la nota media de dos o más de los tres candidatos humanos de la prueba de Turing.
>
> Se considerará que el ordenador ha superado la prueba de Turing si supera tanto la prueba para determinar si es humano dentro de la prueba de Turing como la prueba de clasificación.
>
> Si un ordenador supera la prueba de Turing según los criterios mencionados antes de que termine el año 2029, entonces Ray Kurzweil ganará la apuesta. En caso contrario, ganará la apuesta Mitchell Kapor.[92]

Menudas exigencias. Eugene Goostman no tendría ninguna posibilidad. Yo no tendría más remedio que estar de acuerdo (con todas las cautelas) con esta valoración de Kurzweil: «En mi opinión, no hay trucos ni algoritmos más sencillos (es decir, métodos más sencillos que aquellos en los que se apoya la inteligencia humana) que permitan a una máquina superar una prueba de Turing debidamente diseñada si no posee una inteligencia de nivel completamente humano».[93]

Además de exponer las reglas de su apuesta a largo plazo, Kapor y Kurzweil escribieron sendos ensayos en los que explicaban los motivos por los que cada uno creía que iba a ganar. El ensayo de Kurzweil resume los argumentos presentes en sus libros: los progresos exponenciales en computación, neurociencia y nanotecnología, que, todos juntos, harán posible la ingeniería inversa del cerebro.

Kapor no se lo cree. Su principal argumento se centra en la influencia que tienen nuestro cuerpo físico (humano) y nuestras emociones en nuestra cognición. «La percepción del entorno y la interacción [física] con él contribuyen a partes iguales con la cognición a formar la experiencia […]. [Las emociones] delimitan y dan forma a lo pensable».[94] Kapor afirma que sin el equivalente de un cuerpo humano, con todo lo que implica, una máquina nunca podrá aprender todo lo necesario para superar la estricta prueba de Turing suya y de Kurzweil.

> Afirmo que el modo fundamental de aprendizaje de los seres humanos es a través de la experiencia. El aprendizaje a través de los libros es una capa que se superpone. […] Si el conocimiento humano, en especial el conocimiento sobre la experiencia, es en gran parte tácito, es decir, nunca se expresa de forma directa y explícita, no se encontrará en los libros, y el enfoque de Kurzweil sobre la adquisición de conocimientos fracasará. […] El problema no está en lo que el ordenador sabe, sino en lo que no sabe ni puede saber.[95]

Kurzweil responde que está de acuerdo con Kapor sobre el papel del aprendizaje experimental, el conocimiento tácito y las emociones, pero cree que antes de la década de 2030 la realidad virtual será «totalmente realista», [96] hasta el punto de poder reproducir las experiencias físicas necesarias para educar a una inteligencia artificial en desarrollo. (Bienvenidos a Matrix). Además, esta inteligencia artificial tendrá un cerebro artificial creado con ingeniería inversa, uno de cuyos elementos fundamentales será la emoción.

¿Ve usted las predicciones de Kurzweil con escepticismo, como Kapor? Kurzweil dice que es porque no entiende los exponenciales. «En general, mi principal discrepancia con los críticos es que dicen que no valoro

suficientemente la complejidad de la ingeniería inversa del cerebro humano ni la complejidad de la biología. Pero no creo que yo esté minusvalorando el problema. Creo que ellos están minusvalorando el poder del crecimiento exponencial».[97]

Los que dudan de Kurzweil señalan que este argumento tiene un par de fallos. Es cierto que los ordenadores han tenido un progreso exponencial en las últimas cinco décadas, pero hay muchas razones para pensar que esta tendencia no va a continuar en el futuro. (Kurzweil, por supuesto, no está de acuerdo). Pero, sobre todo, los programas informáticos no han tenido el mismo progreso exponencial; es difícil defender que el *software* actual es exponencialmente más sofisticado, más parecido al cerebro, que el *software* de hace cincuenta años, o incluso que alguna vez haya habido esa tendencia. También son muy discutidas las afirmaciones de Kurzweil sobre las tendencias exponenciales de la neurociencia y la realidad virtual.

Pero como señalan los partidarios de la Singularidad, a veces es difícil ver una tendencia exponencial si estamos dentro de ella. En una curva exponencial como las de la figura 5, Kurzweil y sus seguidores imaginan que estamos en ese punto en el que la curva empieza a crecer poco a poco y nos parece un progreso gradual, pero es engañoso: en realidad, el crecimiento está a punto de dispararse.

¿La primavera actual de la IA es, como aseguran muchos, el primer anuncio de una próxima explosión? ¿O no es más que un punto intermedio en una curva de crecimiento lento y gradual que no desembocará en una IA de nivel humano hasta dentro de un siglo? ¿O es otra burbuja de IA, a la que pronto seguirá otro invierno de IA?

Para orientarnos mejor sobre estas dudas, debemos examinar con detalle algunas de las aptitudes fundamentales que hacen distinta la inteligencia humana, como la percepción, el lenguaje, la toma de decisiones, el razonamiento basado en el sentido común y el aprendizaje. En los próximos

capítulos veremos hasta dónde la IA ha sido capaz de absorber estas aptitudes y valoraremos las perspectivas para 2029 y más adelante.

---

[49] Q. V. Le *et al.*, «Building High-Level Features Using Large-Scale Unsupervised Learning», en *Proceedings of the International Conference on Machine Learning* (2012), pp. 507-514.

[50] P. Hoffman, «Retooling Machine and Man for Next Big Chess Faceoff», *The New York Times*, 21 de enero de 2003.

[51] D. L. McClain, «Chess Player Says Opponent Behaved Suspiciously», *The New York Times*, 28 de septiembre de 2006.

[52] Citado en M. Y. Vardi, «Artificial Intelligence: Past and Future», *Communications of the Association for Computing Machinery* 55, n.º 1 (2012), p. 5.

[53] K. Kelly, «The Three Breakthroughs That Have Finally Unleashed AI on the World», *Wired*, 27 de octubre de 2014.

[54] J. Despres, «Scenario: Shane Legg», *Future*, consultado el 4 de diciembre de 2018, future.wikia.com/wiki/Scenario:ShaneLegg.

[55] Citado en H. McCracken, «Inside Mark Zuckerberg's Bold Plan for the Future of Facebook», *Fast Company*, 16 de noviembre de 2015, www.fastcompany.com/ 3052885/mark-zuckerberg-facebook.

[56] V. C. Müller y N. Bostrom, «Future Progress in Artificial Intelligence: A Survey of Expert Opinion», en *Fundamental Issues of Artificial Intelligence*, ed. de V. C. Müller, Cham, Suiza: Springer International, 2016, pp. 555-572.

[57] M. Loukides y B. Lorica, «What Is Artificial Intelligence?», *O'Reilly*, 20 de junio de 2016, www.oreilly.com/ideas/what-is-artificial-intelligence.

[58] S. Pinker, «Thinking Does Not Imply Subjugating», en *What to Think About Machines That Think*, ed. de J. Brockman, Nueva York: Harper Perennial, 2015, pp. 5-8.

[59] A. M. Turing, «Computing Machinery and Intelligence», *Mind* 59, n.º 236 (1950), pp. 433-460.

[60] J. R. Searle, «Minds, Brains, and Programs», *Behavioral and Brain Sciences* 3, (1980), pp. 417-424.

[61] J. R. Searle, *Mind: A Brief Introduction*, Oxford: Oxford University Press, 2004, p. 66.

[62] Los términos *IA fuerte* e *IA débil* también se han utilizado para referirse a algo más parecido a *IA general* e *IA estrecha*. Así es como los utiliza Ray Kurzweil, pero difiere del significado original de Searle.

[63] El artículo de Searle se reproduce en D. R. Hofstadter y D. C. Dennett, *The Mind's I: Fantasies and Reflections on Self and Soul* (Nueva York: Basic Books, 1981), junto con un convincente contraargumento de Hofstadter.

[64] S. Aaronson, *Quantum Computing Since Democritus*, Cambridge, R.U.: Cambridge University Press, 2013, p. 33.

[65] «Turing Test Transcripts Reveal How Chatbot 'Eugene' Duped the Judges», Coventry University, 30 de junio de 2015, www.coventry.ac.uk/primary-news/turingtest-transcripts-reveal-how-chatbot-eugene-duped-the-judges/.

[66] «Turing Test Success Marks Milestone in Computing History», University of Reading, 8 de junio de 2014, www.reading.ac.uk/news-and-events/releases/PR583836.aspx.

[67] R. Kurzweil, *The Singularity Is Near: When Humans Transcend Biology,* Nueva York: Viking Press, 2005, p. 7.

[68] *Ibid.*, pp. 22-23.

[69] I. J. Good, «Speculations Concerning the First Ultraintelligent Machine», *Advances in Computers* 6 (1966), pp. 31-88.

[70] V. Vinge, «First Word», *Omni*, enero de 1983.

[71] Kurzweil, *Singularity Is Near*, pp. 241, 317, 198-199.

[72] B. Wang, «Ray Kurzweil Responds to the Issue of Accuracy of His Predictions», *Next Big Future*, 19 de enero de 2010, www.nextbigfuture.com/2010/01/ray-kurzweil-responds-to-issue-of.html.

[73] D. Hochman, «Reinvent Yourself: The Playboy Interview with Ray Kurzweil», *Playboy*, 19 de abril de 2016, www.playboy.com/articles/playboy-interview-ray-kurzweil.

[74] Kurzweil, *Singularity Is Near*, p. 136.

[75] A. Kreye, «A John Henry Moment», en Brockman, *What to Think About Machines That Think*, pp. 394-396.

[76] Kurzweil, *Singularity Is Near*, p. 494.

[77] R. Kurzweil, «A Wager on the Turing Test: Why I Think I Will Win», Kurzweil AI, 9 de abril de 2002, www.kurzweilai.net/a-wager-on-the-turing-test-why-i-think-i-will-win.

[78] *Ibid.*

[79] *Ibid.*

[80] *Ibid.*

[81] M. Dowd, «Elon Musk's Billion-Dollar Crusade to Stop the A.I. Apocalypse», *Vanity Fair*, 26 de marzo de 2017.

[82] L. Grossman, «2045: The Year Man Becomes Immortal», *Time*, 10 de febrero de 2011.

[83] Del sitio web de la Singularity University, consultado el 4 de diciembre de 2018, su.org/about/.

[84] Kurzweil, *Singularity Is Near*, p. 316.

[85] R. Kurzweil, *The Age of Spiritual Machines: When Computers Exceed Human Intelligence*, Nueva York: Viking Press, 1999, p. 170 [trad. cast.: *La era de las máquinas individuales: cuando los ordenadores superen la mente humana*, Barcelona: Planeta, 1999].

[86] D. R. Hofstadter, «Moore's Law, Artificial Evolution, and the Fate of Humanity», en *Perspectives on Adaptation in Natural and Artificial Systems*, ed. de L. Booker *et al.*, Nueva York: Oxford University Press, 2005, p. 181.

[87] Todas estas citas proceden de Kurzweil, *Age of Spiritual Machines*, pp. 169-170.

[88] Hofstadter, «Moore's Law, Artificial Evolution, and the Fate of Humanity», p. 182.

[89] Del sitio web de Long Bets: longbets.org/about.

[90] Del sitio web de Longs Bet 1: longbets.org/1/#adjudication_terms.

[91] *Ibid.*

[92] *Ibid.*

[93] Kurzweil, «Wager on the Turing Test».

[94] M. Kapor, «Why I Think I Will Win», Kurzweil AI, 9 de abril de 2002, http://www.kurzweilai.net/why-i-think-i-will-win.

[95] *Ibid.*

[96] R. Kurzweil, prólogo a *Virtual Humans*, de P. M. Plantec, Nueva York: AMACOM, 2004.

[97] Citado en Grossman, «2045».

PARTE II

MIRAR Y VER

04

Quién, qué, cuándo, dónde, por qué

Mire la foto de la figura 6 (en la página siguiente) y diga qué ve. Una mujer acariciando a un perro. Una soldado acariciando a un perro. Una soldado que acaba de volver de la guerra a la que su perro recibe con flores y un globo en el que se lee «Bienvenida a casa». La cara de la soldado muestra su mezcla de emociones. El perro mueve alegremente la cola.

¿Cuándo se hizo esta foto? Probablemente en los últimos diez años. ¿Dónde? Probablemente en un aeropuerto. ¿Por qué acaricia la soldado al perro? Probablemente ha estado fuera mucho tiempo, ha vivido muchas cosas, buenas y malas, ha echado mucho de menos al perro y está muy contenta de volver a casa. Quizá el perro es el símbolo de todo lo que significa «casa». ¿Qué pasó justo antes de que se hiciera esta foto? Probablemente la soldado bajó de un avión y recorrió la parte restringida del aeropuerto hasta el lugar de recepción de los viajeros. Su familia o sus amigos la acogieron con abrazos, le dieron las flores y el globo, y soltaron la correa del perro. El perro se acercó a la soldado, que dejó todo lo que llevaba y se arrodilló, con cuidado de sujetar la cuerda del globo bajo la rodilla para evitar que saliera volando. ¿Qué pasará después? Probablemente se levantará, se secará las lágrimas, cogerá las flores, el

globo, el ordenador portátil y la correa del perro, y se irá con él y la familia o los amigos a la zona de recogida de equipajes.

Cuando miramos esta imagen, lo primero que vemos son trazos de tinta en una página (o píxeles en una pantalla). Pero nuestros ojos y nuestro cerebro son capaces de asimilar esa información en bruto y, en pocos segundos, transformarla en una historia detallada que incluye seres vivos, objetos, relaciones, lugares, emociones, motivos y acciones pasadas y futuras. Miramos, vemos y entendemos. Y, sobre todo, sabemos qué no tener en cuenta. Hay muchos aspectos de la foto que no son estrictamente necesarios para entender la historia que nos cuenta: el dibujo de la alfombra, las correas que cuelgan de la mochila de la soldado, el silbato enganchado en la correa, los pasadores que lleva en el pelo.



Figura 6. ¿Qué ve en esta foto?

Los seres humanos procesamos ese enorme volumen de información en muy poco tiempo sin ser apenas conscientes de lo que hacemos o de cómo lo hacemos. Salvo para una persona ciega de nacimiento, el procesamiento visual, en varios grados de abstracción, domina el cerebro.

Desde luego, la capacidad de describir así el contenido de una fotografía (o un vídeo, o una retransmisión en directo desde una cámara) sería una de las primeras cosas que exigiríamos de una IA general de nivel humano.

Las cosas fáciles son difíciles (sobre todo en la visión)

Los investigadores sobre IA se dedican desde los años cincuenta a tratar de conseguir que los ordenadores encuentren sentido a los datos visuales. En los primeros tiempos de la IA, ese era un objetivo que parecía relativamente sencillo. En 1966, Marvin Minsky y Seymour Papert —los profesores del MIT promotores de la IA simbólica que vimos en el capítulo 1— propusieron el Proyecto Visión de Verano, en el que escogerían a estudiantes universitarios para trabajar en «la construcción de una parte significativa de un sistema visual».[98] En palabras de un historiador de la IA, «Minsky contrató a un alumno de primer curso y le asignó un problema para que lo resolviera durante el verano: conectar una cámara de televisión a un ordenador y conseguir que este describiera lo que veía».[99]

El alumno no llegó muy lejos. Y aunque el subcampo de la IA llamado visión por ordenador ha progresado enormemente en las décadas transcurridas desde aquel proyecto de verano, todavía parece imposible un programa capaz de mirar y describir fotografías de la misma forma que lo hacen los seres humanos. Resulta que la visión —tanto mirar como ver—es una de las cosas más difíciles de todas las «fáciles».

Una condición previa indispensable para describir los estímulos visuales es saber reconocer objetos, es decir, identificar un grupo concreto de píxeles en una imagen como una categoría de objeto determinada, por ejemplo «mujer», «perro», «globo» u «ordenador portátil». Reconocer objetos suele ser una tarea tan inmediata y sencilla para los humanos que no parecía que fuera a ser un problema especialmente difícil para los ordenadores, hasta que los investigadores de IA intentaron que lo hicieran.

¿Por qué es tan difícil el reconocimiento de objetos? Pensemos, por ejemplo, en el problema de conseguir que un programa informático reconozca perros en fotografías. La figura 7 (en la página siguiente) muestra algunas de las dificultades. Si los datos no son más que los píxeles de la imagen, el programa tiene que averiguar primero cuáles son píxeles «de perro» y cuáles son píxeles «de no perro» (por ejemplo, fondo, sombras, otros objetos). Además, los perros pueden ser muy distintos: pueden tener diferente color, forma y tamaño; pueden estar mirando en distintas direcciones; la luz puede variar de forma considerable entre unas imágenes y otras; puede haber partes del perro ocultas por otros objetos (por ejemplo, vallas, personas). Los «píxeles de perro» pueden parecerse mucho a los «píxeles de gato» u otros animales. En ciertas condiciones de luz, incluso una nube del cielo puede parecerse mucho a un perro.



Figura 7. Reconocimiento de objetos: fácil para los humanos, difícil para los ordenadores.

La investigación de la visión por ordenador se ha enfrentado a estos y otros problemas desde los años cincuenta. Hasta hace poco, una de las principales tareas de los investigadores sobre este tema era desarrollar algoritmos especializados en procesamiento de imágenes capaces de identificar «características invariables» de los objetos que pudieran utilizarse para reconocerlos a pesar de las dificultades que he enumerado. Pero incluso con un procesamiento de imágenes avanzado, la capacidad de los programas de reconocimiento de objetos seguía siendo muy inferior a la de los humanos.

La revolución del aprendizaje profundo

La capacidad de las máquinas para reconocer objetos en imágenes y vídeos experimentó un salto cualitativo en la década de 2010 gracias a los avances en el área denominada del aprendizaje profundo.

«Aprendizaje profundo» se refiere sencillamente a los métodos para entrenar las «redes neuronales profundas», que, a su vez, designan unas redes neuronales con más de una capa oculta. Recordemos que las capas ocultas son las capas de una red neuronal situadas entre la entrada y la salida. La profundidad de una red es el número de capas ocultas que tiene: una red «superficial» —como la que vimos en el capítulo 2— no tiene más que una capa oculta; una red «profunda» tiene más. Conviene subrayar esta definición: «profundo», en el aprendizaje profundo, no se refiere a la complejidad de lo que se aprende; solo se refiere a la profundidad de la red que se entrena, medida en número de capas.

Hace ya varias décadas que se investiga sobre redes neuronales profundas. Lo que ha hecho que ahora sean una revolución es su increíble éxito reciente en muchas tareas de IA. Los investigadores han descubierto que las redes profundas que mejor funcionan son aquellas cuya estructura imita partes del sistema visual del cerebro. Las redes neuronales multicapa «tradicionales» que describí en el capítulo 2 se inspiran en el cerebro, pero su estructura es muy distinta. En cambio, las redes neuronales que dominan el aprendizaje profundo están construidas directamente basándose en los descubrimientos de la neurociencia.

El cerebro, el neocognitrón y las redes neuronales convolucionales

Más o menos en la misma época en que Minsky y Papert proponían su Proyecto Visión de Verano, había dos neurocientíficos inmersos en un estudio que se prolongó durante varias décadas y que iba a modificar por completo nuestra comprensión de la visión —y en especial el

reconocimiento de objetos— en el cerebro. Posteriormente, David Hubel y Torsten Wiesel recibieron el Premio Nobel por sus descubrimientos sobre la organización jerárquica de los sistemas de visión de los felinos y los primates (incluidos los humanos), y por su explicación de cómo el sistema visual transforma la luz que llega a la retina en información sobre lo que está en la escena.

Los hallazgos de Hubel y Wiesel inspiraron a un ingeniero japonés llamado Kunihiko Fukushima, que en los años setenta desarrolló una de las primeras redes neuronales profundas, el cognitrón, y su sucesora, el neocognitrón. En sus artículos, Fukushima plasmaba algunos éxitos logrados en el entrenamiento del neocognitrón para reconocer cifras escritas a mano (como los que mostré en el capítulo 1), pero los métodos específicos de aprendizaje que había utilizado no parecían servir para tareas visuales más complejas.[100] Aun así, el neocognitrón fue una fuente de inspiración importante para otras formas posteriores de enfocar las redes neuronales profundas, incluida la más influyente y utilizada en la actualidad: las redes neuronales convolucionales o (para la mayoría de la gente del sector, ConvNet o CNN, por sus siglas en inglés).

Las ConvNet son el auténtico motor de la revolución que atraviesa hoy el aprendizaje profundo en visión por ordenador y en otras áreas. Aunque muchos aseguran que las ConvNet son la gran novedad de la IA, en realidad no son tan nuevas: las propuso por primera vez en los años ochenta el informático francés Yann LeCun, que se había inspirado en el neocognitrón de Fukushima.

Voy a detenerme a describir cómo funcionan las ConvNet, porque saberlo es crucial para hacernos una idea del estado actual y los límites de la visión por ordenador y de muchas otras cosas relacionadas con la IA.

Reconocimiento de objetos en el cerebro y en las ConvNet

El diseño de las ConvNet, como el del neocognitrón, se basa en varios hallazgos fundamentales sobre el sistema visual del cerebro que hicieron Hubel y Wiesel en los años cincuenta y sesenta. Cuando los ojos se fijan en una escena, lo que reciben es la luz de distintas longitudes de onda que reflejan los objetos y las superficies de esa escena. La luz que llega a los ojos activa las células de la retina, que es básicamente una red de neuronas en la parte posterior del ojo. Las neuronas comunican su activación al cerebro a través de los nervios ópticos y así activan, a su vez, las neuronas de la corteza visual, que está en la parte posterior de la cabeza (figura 8). La corteza visual está organizada aproximadamente como una serie jerárquica de capas de neuronas, como los pisos de una tarta nupcial, y las neuronas de cada capa comunican su activación a las neuronas de la capa siguiente.

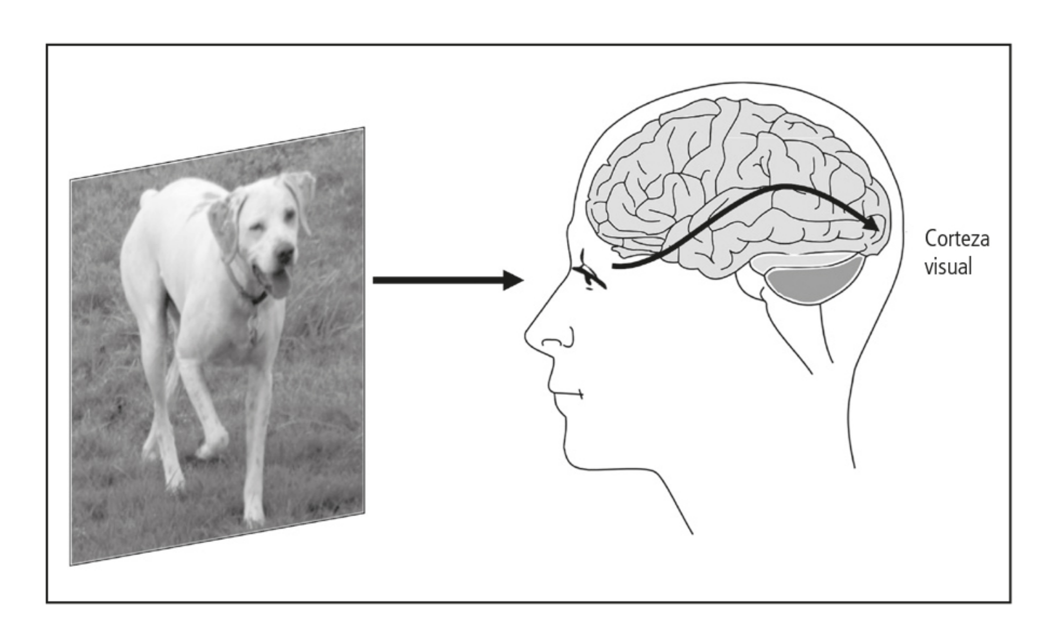


Figura 8. Vía de entrada óptica de los ojos a la corteza visual.

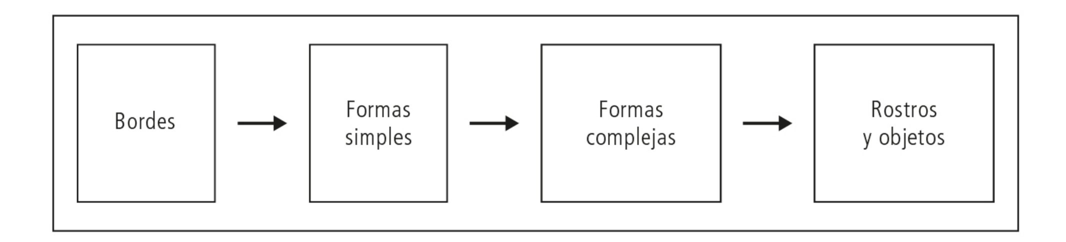


Figura 9. Esquema de las características visuales detectadas por las neuronas en las diferentes capas de la corteza visual.

Hubel y Wiesel encontraron pruebas de que las neuronas de las distintas capas de esta jerarquía actúan como «detectoras» que reaccionan ante los elementos cada vez más complejos que aparecen en la escena visual, como muestra la figura 9: las neuronas de las primeras capas se activan (es decir, se disparan a mayor velocidad) en respuesta a los bordes; su activación alimenta las capas de neuronas que reaccionan ante formas simples compuestas por esos bordes; y así sucesivamente, hasta llegar a formas más complejas y, por último, a objetos enteros y rostros concretos. Obsérvese que las flechas de la figura 9 indican un flujo de información ascendente (o hacia delante), que representa las conexiones desde las capas inferiores hacia las superiores (en la figura, de izquierda a derecha). Es importante señalar que en la corteza visual también se produce un flujo de información descendente o hacia atrás (de las capas superiores a las inferiores); de hecho, hay aproximadamente diez veces más conexiones descendentes que ascendentes. Sin embargo, los neurocientíficos no comprenden del todo la función de estas conexiones hacia atrás, aunque se sabe que los conocimientos y las expectativas previos, seguramente almacenados en capas cerebrales superiores, influyen mucho en lo que percibimos.

Al igual que la estructura jerárquica de transmisión hacia delante ilustrada en la figura 9, una ConvNet está formada por una secuencia de capas de neuronas simuladas, que llamaré de nuevo «unidades». Las unidades de cada capa proporcionan el estímulo a las unidades de la capa siguiente. Como ocurre en la red neuronal que describí en el capítulo 2, cuando una ConvNet procesa una imagen, cada unidad adquiere un valor de activación determinado, un número real que se calcula a partir de las entradas de la unidad con sus respectivos pesos.

Para ser más concretos, imaginemos una ConvNet hipotética, con cuatro capas más un «módulo de clasificación», que queremos entrenar para

reconocer perros y gatos en imágenes. Supongamos, para simplificar, que cada imagen de entrada representa exactamente un perro o un gato. La figura 10 ilustra la estructura de nuestra ConvNet. Es un poco complicada, así que vamos a repasarla con cuidado, paso a paso, para explicar cómo funciona.

Entrada y salida

La entrada o el estímulo de nuestra ConvNet es una imagen, es decir, una matriz de números que corresponden al brillo y el color de los píxeles de la imagen.[101] La salida final que emite nuestra ConvNet es la confianza de la red (del 0 por ciento al 100 por ciento) en cada categoría: «perro» y «gato». Nuestro objetivo es que la red aprenda a emitir una gran confianza en la categoría acertada y una seguridad escasa sobre la otra categoría. Así, la red aprenderá qué conjunto de características de la imagen de entrada es más útil para esta tarea.

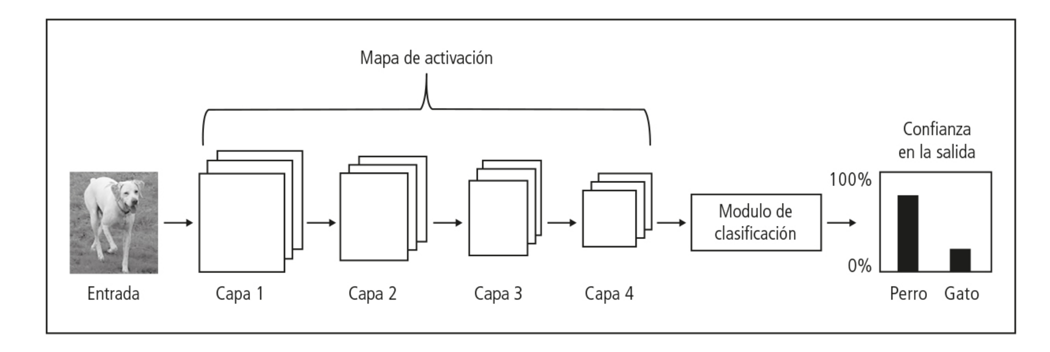


Figura 10. Ilustración de una red neuronal convolucional (ConvNet) de cuatro capas diseñada para reconocer perros y gatos en fotos.

Mapas de activación

Obsérvese en la figura 10 que cada capa de la red está representada por un conjunto de tres rectángulos superpuestos. Estos rectángulos representan mapas de activación, que se inspiran en «mapas» similares encontrados en el sistema visual del cerebro. Hubel y Wiesel descubrieron que las neuronas de las capas inferiores de la corteza visual están dispuestas físicamente de

tal modo que forman más o menos una cuadrícula, en la que cada neurona reacciona a una pequeña zona correspondiente del campo visual. Imaginemos que estamos sobrevolando Los Ángeles de noche en avión y hacemos una foto; las luces que se ven en la foto forman un mapa aproximado de los elementos de la ciudad iluminada. De la misma forma, las activaciones de las neuronas de cada capa cuadriculada de la corteza visual forman un mapa aproximado de los elementos principales de la escena observada. Ahora imaginemos que tenemos una cámara especial, capaz de hacer fotos separadas de las luces domésticas, las luces de los edificios y las luces de los coches. Eso es más o menos lo que hace la corteza visual: cada elemento visual importante tiene su propio mapa neuronal. La combinación de estos mapas contribuye de forma esencial a nuestra percepción de una escena.

Las unidades de una ConvNet, como las neuronas de la corteza visual, actúan como detectoras de elementos visuales importantes; cada unidad busca su elemento correspondiente en una parte concreta del campo visual. Y como ocurre en la corteza visual (más o menos), cada capa de una ConvNet está compuesta por varias cuadrículas de unidades y cada cuadrícula forma un mapa de activación para un elemento visual específico.

¿Qué elementos visuales deben detectar las unidades de una ConvNet? Fijémonos primero en el cerebro. Hubel y Wiesel descubrieron que las neuronas de las capas inferiores de la corteza visual sirven para detectar bordes, teniendo en cuenta que «borde» se refiere al límite entre dos regiones distintas de la imagen. Cada neurona recibe un estímulo correspondiente a una pequeña región concreta de la escena visual; esta región se denomina campo receptivo de la neurona. La neurona se activa (es decir, empieza a emitir a más velocidad) solo si su campo receptivo contiene un tipo concreto de borde.

De hecho, estas neuronas son muy específicas en cuanto al tipo de borde al que reaccionan. Algunas neuronas solo se activan cuando en su campo

receptivo hay un borde vertical; otras solo responden a un borde horizontal; otras solo se activan cuando hay bordes en otros ángulos concretos. Uno de los descubrimientos más importantes de Hubel y Wiesel fue que cada pequeña región del campo visual corresponde a los campos receptivos de muchas neuronas «detectoras de bordes» diferentes. Es decir, en un nivel bajo de procesamiento visual, las neuronas averiguan qué orientación tienen los bordes en cada parte de la escena que observamos. Las neuronas detectoras de bordes lo comunican a las capas superiores de la corteza visual, cuyas neuronas parece que detectan formas, objetos y rostros concretos.[102]

Del mismo modo, la primera capa de nuestra ConvNet hipotética está formada por unidades detectoras de bordes. La figura 11 muestra una vista más detallada de la primera capa de nuestra ConvNet. Esta capa está compuesta por tres mapas de activación, cada uno de los cuales es una cuadrícula de unidades. Cada unidad de un mapa corresponde a la posición análoga en la imagen de entrada, y cada unidad recibe su estímulo de una pequeña región alrededor de esa posición: ese es su campo receptivo. (Los campos receptivos de unidades vecinas suelen solaparse). Cada unidad de cada mapa calcula un valor de activación que mide el grado de «coincidencia» de la región con la orientación de borde preferida de la unidad; por ejemplo, vertical, horizontal o con diversos grados de inclinación.

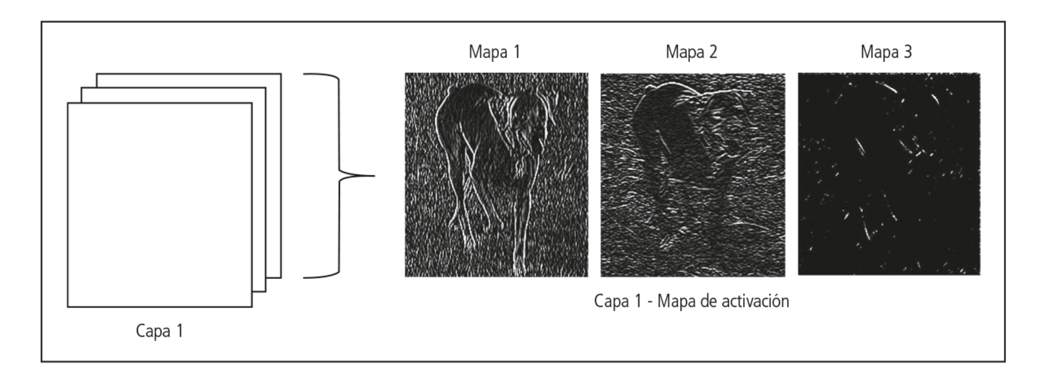


Figura 11. Mapas de activación en la primera capa de nuestra ConvNet

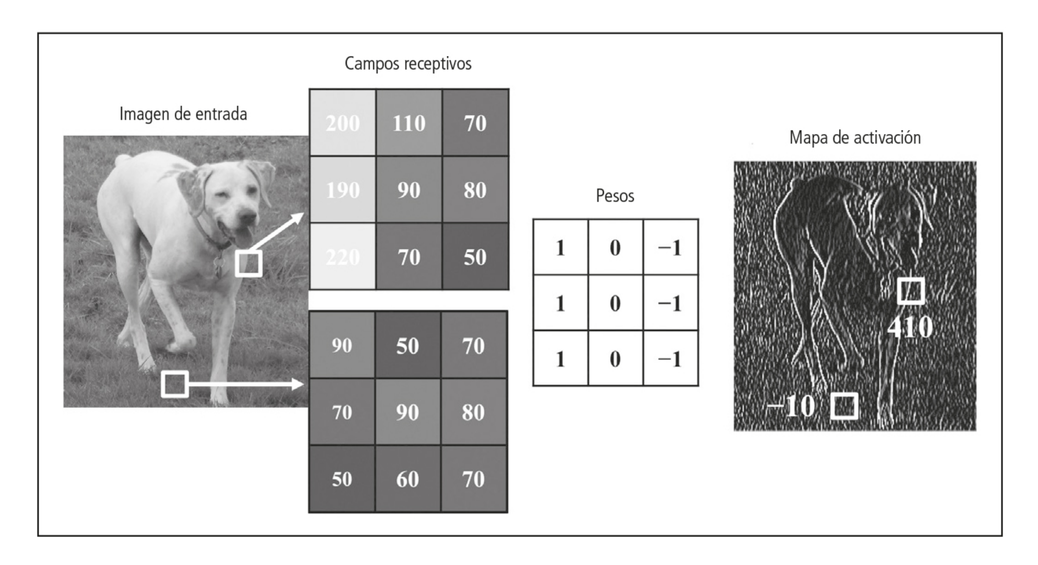


Figura 12. Ilustración de cómo se utilizan las convoluciones para detectar bordes verticales. Por ejemplo, una convolución del campo receptivo superior con los pesos es (200 × 1) + (110 × 0) + (70 × −1) + (190 × 1) + (90 × 0) + + (80 × −1) + (220 × 1) + (70 × 0) + (50 × −1) = 410.

La figura 12 muestra en detalle cómo calculan sus activaciones las unidades del mapa 1, las que detectan los bordes verticales. Los pequeños cuadrados blancos de la imagen que entra representan los campos receptivos de dos unidades diferentes. Al ampliarlos, los fragmentos de imagen dentro de esos campos receptivos aparecen como matrices de valores de píxeles. Aquí, para simplificar, he representado cada fragmento como un conjunto de 3 × 3 píxeles (los valores, por convenio, van de 0 a 255: cuanto más claro es el píxel, mayor es el valor). Cada unidad recibe como entrada los valores de los píxeles de su campo receptivo. Después, la unidad multiplica cada entrada por su peso y suma los resultados para activar la unidad.

Los pesos que aparecen en la figura 12 están pensados para generar una activación positiva elevada cuando hay un borde vertical entre claro y oscuro en el campo receptivo (es decir, un gran contraste entre los lados izquierdo y derecho del fragmento que llega a la neurona). El campo receptivo superior contiene un borde vertical: la piel clara del perro al lado de la hierba, más oscura. Eso se refleja en el valor elevado de activación

(cuatrocientos diez). El campo receptivo inferior no contiene un borde así, solo hierba oscura, y la activación (menos diez) está más cerca de cero. Obsérvese que un borde vertical entre oscuro y claro producirá un valor negativo «elevado» (es decir, un valor negativo alejado de cero).

Este cálculo —multiplicar cada valor de un campo receptivo por su peso correspondiente y sumar los resultados— se denomina convolución. De ahí el nombre de «red neuronal convolucional». Antes he dicho que, en una ConvNet, un mapa de activación es una cuadrícula de unidades que corresponden a campos receptivos situados en toda la imagen. Cada unidad de un mapa de activación determinado utiliza los mismos pesos para calcular una convolución con su campo receptivo; imaginemos la imagen de entrada en la que el cuadrado blanco se desliza a lo largo de cada fragmento de la imagen.[103] El resultado es el mapa de activación de la figura 12: el píxel central del campo receptivo de una unidad es de color blanco para las activaciones positivas y negativas altas, y de color más oscuro para las activaciones cercanas a cero. Se puede ver que las zonas blancas resaltan las posiciones en las que hay bordes verticales. Los mapas 2 y 3 de la figura 11 se crearon del mismo modo, pero con pesos que resaltan los bordes horizontales e inclinados, respectivamente. Todos juntos, los mapas de las unidades detectoras de bordes de la primera capa proporcionan a la ConvNet una representación de la imagen de entrada como una serie de bordes orientados en diferentes regiones, algo parecido a lo que produciría un programa de detección de bordes.

Detengámonos un instante a hablar de la palabra *mapa*. En el lenguaje cotidiano, *mapa* es la representación espacial de un área geográfica, como una ciudad. Un mapa de París, por ejemplo, muestra un elemento concreto de la ciudad —el trazado de calles, avenidas y callejones—, pero no incluye otros muchos elementos como los edificios, las viviendas, las farolas, los cubos de basura, los árboles frutales o los estanques. Otros mapas se fijan en otras características; hay mapas que destacan los carriles bici de París,

los restaurantes vegetarianos, los parques en los que se admiten perros. Sean cuales sean nuestros intereses, seguro que hay un mapa que nos muestra dónde satisfacerlos. Para explicar París a un amigo que nunca ha estado allí, una forma original de hacerlo puede ser enseñarle esa colección de mapas de «intereses especiales».

Una ConvNet, igual que el cerebro, representa la escena visual como una colección de mapas que reflejan los «intereses» específicos de un conjunto de detectores. En mi ejemplo de la figura 11, estos intereses son las diferentes orientaciones de los bordes. Ahora bien, como veremos más adelante, en las ConvNet la propia red aprende cuáles deben ser sus intereses (es decir, los detectores); dependen de la tarea específica para la que se la entrene.

La elaboración de mapas no es exclusiva de la primera capa de nuestra ConvNet. Como puede verse en la figura 10, hay una estructura similar en todas las capas: cada una tiene un conjunto de detectores, y cada uno de ellos crea su propio mapa de activación. Una de las claves del éxito de la ConvNet es que —también como en el cerebro— estos mapas son jerárquicos: las entradas de las unidades de la capa 2 son los mapas de activación de la capa 1, las entradas de las unidades de la capa 3 son los mapas de activación de la capa 2, y así en todas las capas. En nuestra red hipotética, en la que las unidades de la primera capa reaccionan a los bordes, las unidades de la segunda capa serían sensibles a combinaciones específicas de bordes, como las esquinas y las formas en T. Los detectores de la tercera capa serían sensibles a combinaciones de combinaciones de bordes. A medida que se sube en la jerarquía, los detectores son sensibles a características cada vez más complejas, tal como Hubel, Wiesel y otros observaron en el cerebro.

Nuestra ConvNet hipotética tiene cuatro capas, cada una con tres mapas, pero en la realidad estas redes pueden tener muchas más —a veces cientos —, cada una con distintas cantidades de mapas de activación. Determinar

estos y muchos otros aspectos de la estructura de una ConvNet es fundamental para conseguir que estas complejas redes puedan llevar a cabo una tarea determinada. En el capítulo 3, describí la teoría de I. J. Good sobre una futura «explosión de inteligencia» en la que las propias máquinas crearán máquinas cada vez más inteligentes. Todavía no hemos llegado a eso. Por ahora, para conseguir que las ConvNet funcionen bien hace falta mucha creatividad humana.

La clasificación en las ConvNet

Las capas 1 a 4 de nuestra red se denominan capas convolucionales porque cada una hace convoluciones en la capa precedente (y la primera capa hace convoluciones en los datos de entrada). Cuando entra una imagen, cada capa sucesiva hace sus cálculos, hasta que al final, en la cuarta capa, la red ha elaborado un conjunto de mapas de activación para elementos relativamente complejos. Pueden ser ojos, patas, colas o cualquier otra característica que la red haya aprendido que es útil para clasificar los objetos con los que se ha entrenado (en este caso, perros y gatos). Entonces ha llegado el momento de que el módulo de clasificación utilice esos elementos para predecir qué objeto representa la imagen.

En realidad, el módulo de clasificación es una red neuronal tradicional, similar a la que describí en el capítulo 2. Las entradas del módulo de clasificación son los mapas de activación de la capa convolucional superior. La salida del módulo es un conjunto de valores porcentuales, uno por cada categoría posible, que califican la confianza de la red en que la entrada represente una imagen de esa categoría (en este caso, perro o gato).[104]

Como resumen de esta breve explicación de las ConvNet: una ConvNet, inspirada en los hallazgos de Hubel y Wiesel sobre la corteza visual del cerebro, toma una imagen de entrada y la transforma —mediante convoluciones— en un conjunto de mapas de activación con elementos cada vez más complejos. Los elementos de la capa convolucional más alta

se introducen en una red neuronal tradicional (que he denominado módulo de clasificación), que genera porcentajes de confianza sobre las categorías de objetos conocidas por la red. La categoría de objetos sobre la que hay más seguridad es la que la red devuelve como clasificación de la imagen. [105]

¿Quieren hacer experimentos con una ConvNet bien entrenada? No hay más que hacer una foto de un objeto y subirla al motor de búsqueda por imagen de Google.[106] El buscador ejecutará una ConvNet en la imagen y, basándose en los grados de seguridad resultantes (sobre miles de categorías posibles de objetos), nos dirá su «hipótesis más probable» respecto a la imagen.

El entrenamiento de una ConvNet

Nuestra ConvNet hipotética contiene detectores de bordes en su primera capa, pero en las ConvNet del mundo real los detectores de bordes no están integrados. Las ConvNet reales aprenden de los ejemplos de entrenamiento qué elementos hay que detectar en cada capa y cómo ponderar los pesos en el módulo de clasificación para generar una confianza elevada en la respuesta correcta. Y, como en las redes neuronales tradicionales, todos los pesos pueden aprenderse a partir de los datos mediante el mismo algoritmo de retropropagación que describí en el capítulo 2.

Para ser más concretos, he aquí cómo podríamos entrenar a nuestra ConvNet para que identifique una imagen específica, como un perro o un gato. En primer lugar, tendríamos que reunir muchas imágenes de perros y gatos para formar los «datos de entrenamiento». Además, deberíamos crear un archivo que asigne una etiqueta a cada imagen, es decir, «perro» o «gato». (O mejor aún, podríamos seguir el ejemplo de los investigadores sobre la visión por ordenador y contratar a un alumno de posgrado para que haga todo esto por nosotros. En el caso de un estudiante de posgrado, puede reclutar a un alumno de grado. A nadie le gusta encargarse del etiquetado).

El programa de entrenamiento, al principio, asigna valores aleatorios a todos los pesos de la red. Después, el programa empieza a entrenar: suministra a la red las imágenes, una a una; la red hace sus cálculos en cada capa y, al final, emite porcentajes de seguridad para «perro» y «gato». Para cada imagen, el programa de entrenamiento compara esos valores de salida con los valores «correctos»; por ejemplo, si la imagen es un perro, la seguridad de que es un «perro» debe ser del 100 por ciento y la seguridad de que es un «gato» debe ser del 0 por ciento. A continuación, el programa de entrenamiento utiliza el algoritmo de retropropagación para modificar ligeramente los pesos de toda la red, de modo que la próxima vez que se vea esta imagen los grados de seguridad se acerquen más a los valores correctos.

El desarrollo de este procedimiento —introducir la imagen como entrada, calcular el error en la salida y cambiar las ponderaciones— para cada imagen de los datos de entrenamiento es lo que se denomina un ciclo o una «época» de entrenamiento. Se tardan muchas épocas en entrenar una ConvNet, muchos ciclos durante los que la red procesa cada imagen una y otra vez. Al principio, a la red se le dará muy mal distinguir los perros de los gatos, pero poco a poco, a medida que modifique sus pesos durante muchas épocas, lo hará cada vez mejor. Al final llega un momento en el que la red «converge», es decir, los pesos ya no cambian tanto entre una época y otra, y sabe (en principio) reconocer muy bien los perros y los gatos en las imágenes de los datos de entrenamiento. Pero no sabremos si la red lleva a cabo verdaderamente bien esta tarea en general hasta que veamos si puede utilizar lo que ha aprendido para identificar imágenes que no son de sus datos de entrenamiento. Lo verdaderamente interesante es que, aunque no haya un programador que obligue a las ConvNet a aprender a detectar ningún elemento concreto, cuando se entrenan con grandes conjuntos de fotografías del mundo real, parecen aprender una jerarquía de detectores

similar a la que Hubel y Wiesel descubrieron en el sistema visual del cerebro.

En el próximo capítulo narraré el extraordinario ascenso de las ConvNet desde un relativo anonimato hasta el dominio casi absoluto de la visión artificial, una transformación posible gracias a una revolución tecnológica simultánea: la de los «macrodatos».

---

[98] S. A. Papert, «The Summer Vision Project», MIT Artificial Intelligence Group Vision Memo 100 (7 de julio de 1966), dspace.mit.edu/handle/1721.1/6125.

[99] D. Crevier, *AI: The Tumultuous History of the Search for Artificial Intelligence*, Nueva York: Basic Books, 1993, p. 88 [trad. cast.: *Inteligencia artificial*, Madrid: Acento, 1996].

[100] K. Fukushima, «Cognitron: A Self-Organizing Multilayered Neural Network Model», *Biological Cybernetics* 20, n.os 3-4 (1975), pp. 121-36; K. Fukushima, «Neocognitron: A Hierarchical Neural Network Capable of Visual Pattern Recognition», *Neural Networks* 1, n.º 2 (1988), pp. 119-130.

[101] Antes de introducirla en la red, la imagen debe ajustarse a un tamaño fijo, el mismo que el de la primera capa de la red.

[102] La mayoría de las afirmaciones sobre cómo hace el cerebro alguna tarea tienen que ir acompañadas de muchas reservas; lo mismo pasa con la historia que acabo de esbozar. Aunque lo que he dicho es más o menos así, el cerebro es demasiado complejo y los hallazgos que he esbozado no son más que una pequeña parte de la historia de la visión temprana, que en gran parte los científicos todavía no comprenden del todo.

[103] La matriz de pesos asociada a cada mapa de activación se denomina filtro convolucional o núcleo convolucional.

[104] Aquí estoy usando el término *módulo de clasificación* como versión abreviada de lo que normalmente se llama las capas totalmente conectadas de una red convolucional profunda.

[105] Mi descripción de una ConvNet omite muchos detalles. Por ejemplo, para calcular su activación, una unidad en una capa convolucional lleva a cabo una convolución y luego aplica una función de activación no lineal al resultado. Las ConvNet también suelen incluir otros tipos de capas, como las «capas de agrupamiento». Para más detalles, véase I. Goodfellow, Y. Bengio y A. Courville, *Deep Learning*, Cambridge, Mass.: MIT Press, 2016.

[106] En el momento de escribir este libro, se puede acceder al motor de «búsqueda por imágenes» de Google en images.google.com haciendo clic en el pequeño icono de la cámara en el cuadro de búsqueda.

05

ConvNet e ImageNet

Yann LeCun, el inventor de las ConvNet, ha investigado las redes neuronales toda su vida, desde los años ochenta y a través de los inviernos y las primaveras que ha atravesado este campo. Cuando era estudiante de posgrado y posdoctoral, le fascinaban los perceptrones de Rosenblatt y el neocognitrón de Fukushima, pero se dio cuenta de que este último no contaba con un algoritmo de aprendizaje bien supervisado. Junto con otros investigadores (sobre todo su tutor posdoctoral, Geoffrey Hinton), LeCun ayudó a desarrollar ese método de aprendizaje, que es esencialmente la misma forma de retropropagación que se utiliza hoy en día en las ConvNet. [107]

En los años ochenta y noventa, mientras trabajaba en los Laboratorios Bell, LeCun se fijó en el problema del reconocimiento de cifras y letras manuscritas. Combinó ideas del neocognitrón con el algoritmo de retropropagación y creó así algo de nombre parecido: «LeNet», una de las primeras ConvNet. La capacidad de reconocimiento de cifras manuscritas de LeNet hizo que tuviera un gran éxito comercial: en los años noventa y primeros dos mil, el servicio de correos de Estados Unidos la utilizó para identificar de forma automática códigos postales, y el sector bancario para leer los dígitos de los cheques.

El intento de ampliar el uso de LeNet y su sucesora, ConvNet, a tareas de visión más complejas no salió bien. A mediados de los noventa, las redes neuronales empezaron a caer en desgracia en el sector de la IA y empezó a darse preferencia a otros métodos. Pero LeCun, que seguía confiando en las ConvNet, siguió investigando y mejorándolas. Como dijo de él Geoffrey Hinton: «Fue el portador de la antorcha en la época más oscura».[108]

LeCun, Hinton y otros fieles a las redes neuronales creían que unas versiones mejoradas y más amplias de las ConvNet y otras redes profundas dominarían la visión por ordenador si se las podía entrenar con suficientes datos. No dieron su brazo a torcer y siguieron trabajando por su cuenta durante la primera década de este siglo. En 2012, de pronto, la antorcha que portaban los investigadores sobre las ConvNet ocasionó un incendio en los trabajos sobre la visión al ganar una competición de visión por ordenador con un conjunto de datos de imágenes llamado ImageNet.

La construcción de ImageNet

Los investigadores de IA son muy competitivos, así que no es de extrañar que les guste organizar competiciones para impulsar su trabajo. En el campo del reconocimiento visual de objetos, los investigadores llevan mucho tiempo organizando concursos anuales para averiguar qué programa funciona mejor. Cada uno de estos concursos incluye un «conjunto de datos de referencia»: una colección de fotos y unas etiquetas creadas por personas que nombran los objetos de las fotos. Entre 2005 y 2010, la más importante de estas competiciones anuales fue el concurso de clases de objetos visuales PASCAL, en el que en 2010 se presentaron alrededor de quince mil fotografías (descargadas de la web de intercambio de fotos Flickr), con etiquetas creadas por humanos para designar veinte categorías de objetos, entre las que estaban «persona», «perro», «caballo», «oveja», «coche», «bicicleta», «sofá» y «planta en una maceta».

Los programas que competían en la parte de «clasificación» del concurso[109] eran programas de visión por ordenador capaces de recibir como estímulo una fotografía (sin ver la etiqueta que le había asignado un ser humano) y decir después si en la imagen había un objeto de cada una de las veinte categorías.

La competición se desarrollaba de esta forma: los organizadores distribuían las fotografías entre un conjunto de datos de entrenamiento que los concursantes podían utilizar para entrenar sus programas y un conjunto de datos de prueba que no se dejaba ver a los concursantes y que servía para medir el comportamiento de los programas con imágenes ajenas a los datos de entrenamiento. Antes del concurso se proporcionaban los datos de entrenamiento en línea y, cuando se celebraba la competición, los investigadores presentaban los programas que habían entrenado para que los examinaran utilizando los datos secretos de prueba. El programa ganador era el que más acertaba en el reconocimiento de los objetos de las imágenes de prueba.

Las competiciones anuales PASCAL fueron muy importantes y contribuyeron enormemente a hacer avanzar la investigación sobre el reconocimiento de objetos. Durante los años que se celebró el concurso, los programas fueron mejorando poco a poco (curiosamente, las plantas en maceta seguían siendo los objetos más difíciles de identificar). No obstante, algunos investigadores se sentían frustrados por las limitaciones de PASCAL como método para impulsar la visión por ordenador. Los concursantes prestaban demasiada atención a las veinte categorías de objetos específicas de PASCAL y no creaban sistemas que pudieran abarcar la enorme cantidad de categorías de objetos que reconocen los humanos. Además, los datos que se suministraban no contenían fotos suficientes para que los sistemas participantes pudieran aprender todas las variaciones posibles del aspecto de los objetos y fueran capaces de generalizar.

Para avanzar, el campo necesitaba una nueva colección de imágenes de referencia, con muchas más categorías y muchas más fotos. Fei-Fei Li, una joven profesora de Visión por Ordenador de Princeton, estaba especialmente dedicada a conseguirlo. Se enteró por casualidad de que había un proyecto dirigido por otro profesor de Princeton, el psicólogo George Miller, para crear una base de datos de palabras del inglés, ordenadas en una jerarquía que iba de lo más particular a lo más general, con los sinónimos agrupados. Por ejemplo, tomemos la palabra *capuchino*. La base de datos, llamada WordNet, contiene la siguiente información sobre esta palabra (la flecha significa «es un tipo de»):

capuchino ⇒ café ⇒ bebida ⇒ alimento ⇒ sustancia ⇒ entidad física ⇒ entidad

La base de datos también contiene la información de que, por ejemplo, «bebida», «trago» y «potable» son sinónimos, que «bebida» forma parte de otra cadena que incluye «líquido», y así sucesivamente.

WordNet había sido (y sigue siendo) muy utilizada en las investigaciones de psicólogos y lingüistas, así como en sistemas de procesamiento de lenguaje natural de IA, pero Fei-Fei Li tuvo una idea nueva: crear una base de datos de imágenes estructurada con arreglo a los sustantivos de WordNet en la que cada sustantivo estuviera vinculado a un gran número de imágenes que contuvieran ejemplos de ese sustantivo. Así nació la idea de ImageNet.

Li y sus colaboradores no tardaron en recopilar una avalancha de imágenes a base de buscar sustantivos de WordNet en buscadores de imágenes como Flickr y Google Images. Ahora bien, cualquiera que haya utilizado un buscador de imágenes sabe que los resultados no suelen ser nada perfectos. Por ejemplo, si escribimos «macintosh apple» (manzana macintosh) en el cuadro de búsqueda de imágenes de Google, obtendremos no solo fotos de manzanas y ordenadores Mac, sino también de velas con forma de manzana, teléfonos inteligentes, botellas de licor de manzana y

muchas otras cosas irrelevantes. De modo que Li y sus colegas tuvieron que pedir a unas personas que vieran qué imágenes no eran ilustraciones de un sustantivo determinado y las eliminaran. Al principio, los encargados eran sobre todo estudiantes universitarios. Era un trabajo atroz, lento y agotador. Li se dio cuenta de que, a ese ritmo, tardarían noventa años en completar la tarea.[110]

Li y su equipo estudiaron posibles formas de automatizar el proceso, pero, por supuesto, el problema de decidir si una foto es la representación de un nombre concreto es precisamente en lo que consiste el reconocimiento de objetos. Y los ordenadores estaban muy lejos de poder desempeñar la tarea, lo cual era el motivo por el que se había construido ImageNet.

El grupo se encontraba estancado, hasta que Li, por casualidad, se topó con un sitio web de tres años de antigüedad que podía proporcionar el talento humano que necesitaba ImageNet. El sitio web tenía el extraño nombre de Amazon Mechanical Turk.

Mechanical Turk

Según Amazon, su servicio Mechanical Turk es «un mercado para trabajos que requieren inteligencia humana». El servicio pone en contacto a los «solicitantes» —personas que necesitan llevar a cabo una tarea difícil para los ordenadores— con los «trabajadores» —personas dispuestas a prestar su inteligencia humana a la tarea que necesita un solicitante— a cambio de unos pequeños honorarios (por ejemplo, etiquetar los objetos de una foto por diez céntimos cada foto). Se han inscrito cientos de miles de trabajadores de todo el mundo. Mechanical Turk es la plasmación de la frase de Marvin Minsky «Lo fácil es difícil»: se contrata a trabajadores humanos para que hagan las tareas «fáciles» que, hoy en día, son demasiado difíciles para los ordenadores.

El nombre de Turco Mecánico procede de un famoso engaño de la IA del siglo XVIII: el Turco Mecánico original era una «máquina inteligente» que jugaba al ajedrez; en ella se escondía una persona que manejaba una marioneta (el «turco», vestido como un sultán otomano) con la que hacía las jugadas. Parece que engañó a muchos personajes de la época, incluido Napoleón Bonaparte. El servicio de Amazon, aunque no pretende engañar a nadie, es, como el Turco Mecánico original, una «inteligencia artificial *artificial*».[111]

Fei-Fei Li se dio cuenta de que si su grupo pagaba a decenas de miles de trabajadores de Mechanical Turk para que eliminaran las imágenes irrelevantes para cada uno de los términos de WordNet, se podría completar todo el conjunto de datos en unos cuantos años con un coste relativamente bajo. En apenas dos años, se etiquetaron más de tres millones de imágenes con los sustantivos WordNet correspondientes para formar la base de datos ImageNet. Desde luego, Mechanical Turk fue «una bendición» para el proyecto.[112] Los investigadores en IA siguen recurriendo mucho al servicio para crear conjuntos de datos; en la actualidad, las propuestas de becas académicas en IA suelen incluir una partida para «trabajadores de Mechanical Turk».

Las competiciones de ImageNet

En 2010, el proyecto ImageNet puso en marcha el primer Desafío ImageNet de reconocimiento visual a gran escala, con el fin de estimular los avances para dar con algoritmos de reconocimiento de objetos más generales. Compitieron treinta y cinco programas que representaban a investigadores de visión por ordenador del mundo académico y de la industria de todo el planeta. A los concursantes se les dieron unas imágenes de entrenamiento etiquetadas (1,2 millones) y una lista de posibles categorías. La tarea de los programas entrenados consistía en asignar la categoría correcta a cada

imagen de entrada. ImageNet tenía mil categorías posibles, frente a las veinte de PASCAL.

Las mil categorías posibles eran un subconjunto de términos de WordNet elegidos por los organizadores. Las categorías son un grupo de sustantivos aparentemente aleatorio que van desde lo más familiar y corriente («limón», «castillo», «piano de cola»), hasta cosas menos comunes («viaducto», «cangrejo ermitaño», «metrónomo») y muy poco conocidas («galgo escocés», «vuelvepiedras», «mono patas»). Los animales y plantas más desconocidos —al menos, que yo no podría distinguir— constituyen, como mínimo, una décima parte de las mil categorías utilizadas.

Algunas fotografías no contienen más que un objeto; otras contienen muchos, entre ellos el «correcto». Esta ambigüedad hace que un programa pueda aventurar cinco categorías para cada imagen, y si en esa lista está la correcta, se considera que el programa ha acertado con la imagen. Es lo que se denomina la métrica de precisión de las «cinco mejores».

El programa que más puntos obtuvo en 2010 utilizó una máquina de vectores de soporte, el algoritmo de reconocimiento de objetos predominante en aquel entonces, que empleaba cálculos complejos para aprender a asignar una categoría a cada imagen de entrada. Utilizando la métrica de precisión de las cinco mejores, el programa acertó en el 72 por ciento de las 150.000 imágenes de prueba. No está nada mal, aunque eso quiere decir que el programa se equivocó —a pesar de poder hacer cinco conjeturas— en más de cuarenta mil de las imágenes de prueba, lo que deja mucho margen de mejora. Es llamativo que entre los programas con mejor puntuación no hubo ninguna red neuronal.

En la competición del año siguiente, el programa que más puntos consiguió —también con máquinas de vectores de soporte— mejoró de forma respetable pero discreta, puesto que acertó el 74 por ciento de las imágenes de prueba. La mayoría de los expertos esperaban que esta tendencia se mantuviera: que la investigación en visión por ordenador fuera

resolviendo poco a poco los problemas con mejoras graduales en cada competición anual.

Sin embargo, estas expectativas cambiaron drásticamente en el concurso ImageNet de 2012: el programa que ganó obtuvo nada menos que un 85 por ciento de aciertos. Fue una mejora de la precisión de lo más asombrosa. Además, el programa vencedor no utilizó máquinas de vectores de soporte ni ninguno de los demás métodos de visión por ordenador habituales en aquel entonces. Era una red neuronal convolucional. Esta ConvNet concreta se denomina AlexNet, por el nombre de su principal creador, Alex Krizhevsky, entonces estudiante de posgrado en la Universidad de Toronto, bajo la supervisión del eminente investigador de redes neuronales Geoffrey Hinton. Krizhevsky, en colaboración con Hinton y otro alumno, Ilya Sutskever, creó una versión ampliada de LeNet, que había construido Yann LeCun en los años noventa; ahora que había aumentado la potencia de los ordenadores, ya era posible entrenar una red tan grande. AlexNet tenía ocho capas y alrededor de sesenta millones de pesos, cuyos valores se aprendían mediante retropropagación a partir de más de un millón de imágenes de entrenamiento.[113] El grupo de Toronto concibió una serie de métodos astutos para mejorar el entrenamiento de la red y, gracias a ello, un grupo de potentes ordenadores tardó aproximadamente una semana en entrenar AlexNet.

El éxito de AlexNet sacudió el mundillo de la visión por ordenador y de la IA en general, porque hizo ver de pronto a la gente el poder que podían llegar a tener las ConvNet, que la mayoría de los investigadores del sector no habían tomado en serio hasta entonces como un competidor que tener en cuenta. En un artículo de 2015, el periodista Tom Simonite preguntó a Yann LeCun sobre el inesperado triunfo de las ConvNet:

> LeCun recuerda haber visto a la comunidad, que en general había ignorado las redes neuronales, abarrotar la sala en la que los ganadores presentaron un informe sobre sus resultados. «Muchos miembros prestigiosos del sector se dieron de bruces con la realidad allí mismo —cuenta—. Dijeron: «Vale, ahora nos lo creemos. Ya está, habéis ganado».[114]

Casi al mismo tiempo, el grupo de Geoffrey Hinton estaba demostrando también que las redes neuronales profundas, si eran entrenadas con enormes cantidades de datos etiquetados, superaban con mucho los mejores resultados conseguidos hasta entonces en el ámbito del reconocimiento del habla. Los resultados del grupo de Toronto en ImageNet y reconocimiento del habla tuvieron gran repercusión. Un año después, Google adquirió una pequeña empresa creada por Hinton, de forma que este y sus alumnos Krizhevsky y Sutskever se convirtieron en empleados de Google. Con la compra, Google se situó en la vanguardia del aprendizaje profundo.

Poco después, Facebook convenció a Yann LeCun de que dejara su cátedra en la Universidad de Nueva York para dirigir el laboratorio de IA que acababa de crear. Enseguida, todas las grandes tecnológicas (y muchas otras más pequeñas) se apresuraron a contratar a expertos en aprendizaje profundo y estudiantes de posgrado lo antes posible. El aprendizaje profundo se convirtió, aparentemente de la noche a la mañana, en el aspecto más de moda de la IA, y los conocimientos sobre el tema empezaron a garantizar a los informáticos un sueldo sustancioso en Silicon Valley o, mejor aún, financiación de capital riesgo para un número cada vez mayor de nuevas empresas dedicadas al aprendizaje profundo.

El concurso anual de ImageNet empezó a recibir más atención de los medios de comunicación y rápidamente dejó de ser una competición académica amistosa para convertirse en un combate muy publicitado entre empresas tecnológicas que comercializaban la visión por ordenador. Ganar en ImageNet garantizaba el ansiado respeto de los especialistas en visión, además de una publicidad gratuita que podría traducirse en ventas de productos y subidas del precio de las acciones. La presión para crear programas que superasen a los de la competencia quedó clara en un incidente de 2015 en el que se descubrió que la gigantesca empresa china de internet Baidu estaba haciendo trampas. El engaño fue un sutil ejemplo de lo que en el ámbito del aprendizaje automático se llama dragado de datos.

He aquí lo que ocurrió: antes del concurso, a cada equipo que competía en ImageNet se le suministraron imágenes de entrenamiento etiquetadas con las categorías de objetos correctas. También se les dio un gran conjunto de datos de prueba —una colección de imágenes no incluidas en los datos de entrenamiento— sin ninguna etiqueta. Una vez entrenado el programa, el equipo podía ver si su método daba buenos resultados con las imágenes de prueba. De esa forma es posible comprobar si un programa ha aprendido a generalizar (en lugar de, por ejemplo, memorizar las imágenes de entrenamiento y sus etiquetas). Lo único que cuenta es el comportamiento con las imágenes de prueba. La forma que tenía un equipo de averiguar si su programa funcionaba bien en el conjunto de pruebas era ejecutar el programa para cada imagen del conjunto de pruebas, recopilar las cinco mejores conjeturas sobre cada imagen y enviar la lista a un «servidor de la prueba», un ordenador manejado por los organizadores del concurso. El servidor de la prueba comparaba la lista enviada con las respuestas correctas (secretas) y emitía el porcentaje de aciertos.

Cada equipo podía registrarse para hacerse una cuenta en el servidor de la prueba y así obtener en retorno la puntuación que habían conseguido las distintas versiones de sus programas; de esa forma podían publicar (y dar a conocer) sus resultados antes de que se anunciaran los resultados oficiales.

En el aprendizaje automático existe una regla sagrada: «No entrenar con los datos de prueba». Parece de cajón: si incluimos datos de prueba en alguna parte del entrenamiento de nuestro programa, no podremos ver exactamente qué capacidad de generalización tiene. Es como dar a los alumnos las preguntas del examen final antes de que lo hagan. Pero resulta que hay formas sutiles de romper esta regla, de forma voluntaria o involuntaria, para que el comportamiento de nuestro programa parezca mejor de lo que es.

Un método sería enviar las respuestas del programa sobre los datos de prueba al servidor y, en función del resultado, hacer alguna modificación

para luego volver a enviarlo. Se puede repetir varias veces, hasta que tenga los ajustes necesarios para identificar mejor los datos de prueba. Para eso no hace falta ver materialmente las etiquetas de los datos de prueba, pero sí conocer hasta qué punto han sido acertadas las conjeturas y ajustar el programa en consecuencia. Si se hace suficientes veces, puede mejorar mucho el comportamiento del programa con los datos de prueba. Pero al usar los datos de prueba para modificar el programa, ya no se pueden utilizar para saber si el programa es capaz de generalizar. Sería como permitir a unos estudiantes que se presenten muchas veces a un examen final y ponerles cada vez una sola nota que les sirve para intentar mejorar la vez siguiente, hasta que, al final, los alumnos presentan las respuestas que hayan tenido la mejor puntuación. Entonces ya no sirve para saber si los alumnos han aprendido bien la asignatura, sino solo para saber cómo han adaptado sus respuestas a unas preguntas concretas.

Para evitar este tipo de dragado de datos pero dejar que los competidores de ImageNet pudieran seguir comprobando el comportamiento de sus programas, los organizadores establecieron la norma de que cada equipo podía enviar respuestas al servidor de prueba un máximo de dos veces por semana. Así se limitaba la cantidad de valoraciones de retorno que los equipos podrían obtener de esos ensayos.

La gran batalla de ImageNet de 2015 se libró por una fracción de punto porcentual, una diferencia aparentemente nimia pero que podía ser muy lucrativa. A principios de año, un equipo de Baidu anunció un método que conseguía la mayor precisión (los cinco mejores) vista en unos datos de prueba de ImageNet: 94,67 por ciento, para ser exactos. Ahora bien, ese mismo día, un equipo de Microsoft anunció que su método obtenía más precisión: el 95,06 por ciento. A los pocos días, un equipo de Google anunció un método ligeramente diferente que acertaba todavía más: el 95,18 por ciento. Este récord se mantuvo durante unos meses, hasta que Baidu hizo un nuevo anuncio: había mejorado su método y ahora podía presumir

de un porcentaje todavía mejor, el 95,42 por ciento. El equipo de relaciones públicas de la empresa dio gran publicidad al resultado.

Sin embargo, unas semanas después, los organizadores de ImageNet emitieron un escueto comunicado: «Durante el periodo comprendido entre el 28 de noviembre de 2014 y el 13 de mayo de 2015, un equipo de Baidu utilizó al menos treinta cuentas distintas para enviar datos al servidor de pruebas un mínimo de doscientas veces, muy por encima del límite establecido de dos envíos por semana».[115] En resumen, habían descubierto que el equipo de Baidu hacía dragado de datos.

Las doscientas puntuaciones conseguidas de vuelta permitieron al equipo de Baidu determinar qué ajustes necesitaba su programa para acertar más con los datos de prueba y así ganar la importantísima fracción de punto porcentual que daba la victoria. Como castigo, Baidu quedó descalificada para presentar su programa en el concurso de 2015.

Con la esperanza de acallar la mala publicidad, la empresa se disculpó de inmediato y culpó a un empleado que había actuado por su cuenta: «Descubrimos que un jefe de equipo había ordenado a los ingenieros ayudantes que enviaran más de dos propuestas a la semana, lo que infringía las normas vigentes de ImageNet».[116] El empleado, pese a negar que hubiera infringido ninguna norma, fue despedido inmediatamente de la empresa.

Aunque esta historia no es más que una interesante nota a pie de página en la historia general del aprendizaje profundo en visión por ordenador, la cuento para ilustrar hasta qué punto la competición de ImageNet llegó a ser el símbolo fundamental de progreso en visión por ordenador y en IA en general.

Al margen de que se hicieran trampas, los avances en ImageNet continuaron. La última competición se celebró en 2017, con un nivel de acierto del 98 por ciento en las cinco mejores conjeturas. Como comentó un periodista: «Hoy en día, muchos consideran que el tema de ImageNet ha

quedado resuelto»,[117] por lo menos en cuanto a la tarea de clasificación. El sector está avanzando hacia nuevos conjuntos de datos de referencia y nuevos problemas, sobre todo los que integran la visión y el lenguaje.

¿Qué fue lo que permitió a las ConvNet, que parecían estar en un callejón sin salida en los años noventa, pasar a dominar de repente la competición de ImageNet y después la mayor parte de la visión por ordenador en los últimos cinco años? En realidad, el éxito reciente del aprendizaje profundo se debe, más que a los nuevos avances en IA, a la posibilidad de disponer de enormes cantidades de datos (¡gracias, internet!) y dispositivos informáticos muy rápidos. Estos factores, junto con la mejora de los métodos de entrenamiento, permiten entrenar redes de más de cien capas con millones de imágenes en solo unos días.

Al propio Yann LeCun le sorprendió con qué rapidez cambiaron las cosas para su ConvNet: «No es frecuente que una tecnología que lleva veinte o veinticinco años en el mercado, prácticamente sin cambios, acabe siendo la mejor. Es asombroso a qué velocidad la ha adoptado la gente. Nunca había visto nada igual».[118]

La fiebre del oro de las ConvNet

Después de que ImageNet y otros grandes conjuntos de datos proporcionaran a las ConvNet la enorme cantidad de ejemplos de entrenamiento que necesitaban para funcionar bien, las empresas se encontraron de pronto con que podían utilizar la visión por ordenador de formas nunca vistas. En palabras de Blaise Agüera y Arcas, de Google: «Ha sido una especie de fiebre del oro: hemos abordado un problema tras otro con las mismas técnicas».[119] Utilizando ConvNet entrenadas mediante aprendizaje profundo, los motores de búsqueda de imágenes de Google, Microsoft y otros pudieron mejorar enormemente su función de «encontrar imágenes similares». Google ofrecía un sistema de almacenamiento de fotos que permitía etiquetarlas con la descripción de los objetos que

contenían, y el servicio Street View de Google podía identificar y difuminar en sus imágenes direcciones de calles y matrículas. La proliferación de aplicaciones móviles hizo posible que los teléfonos inteligentes reconocieran objetos y rostros sobre la marcha.

Facebook empezó a etiquetar las fotos que subíamos con los nombres de nuestros amigos y registró una patente para clasificar las emociones que delataban las expresiones faciales de las fotos; Twitter desarrolló un filtro capaz de detectar imágenes pornográficas en los tuits; y varias webs de intercambio de fotos y vídeos empezaron a aplicar herramientas para detectar imágenes asociadas a grupos terroristas. Las ConvNet pueden aplicarse en vídeos y utilizarse para que los coches autónomos rastreen la presencia de peatones, para leer los labios y para clasificar el lenguaje corporal. Las ConvNet pueden incluso diagnosticar cáncer de mama y de piel a partir de imágenes médicas, determinar el estadio de la retinopatía diabética y ayudar a los médicos a planificar el tratamiento del cáncer de próstata.

Estos son solo algunos ejemplos de las muchas aplicaciones comerciales que ya funcionan (o pronto funcionarán) gracias a las ConvNet. De hecho, es muy probable que cualquier aplicación moderna de visión por ordenador de las que utilizamos emplee ConvNet. Y es más que probable que la hayan «entrenado previamente» con imágenes de ImageNet para aprender características visuales genéricas antes de «refinarla» para tareas más específicas.

Dado que el intenso entrenamiento que necesitan las ConvNet solo es factible con material informático especializado —normalmente, unidades de procesamiento gráfico (GPU) muy potentes—, no es extraño que el precio de las acciones de NVIDIA Corporation, el fabricante más destacado de GPU, se multiplicara más de un 1.000 por ciento entre 2012 y 2017.

¿Han superado las ConvNet a los humanos en el

reconocimiento de objetos?

A medida que conocía los extraordinarios éxitos de las ConvNet, me preguntaba cuánto se aproximaban a la capacidad humana de reconocer objetos. Un artículo publicado por Baidu en 2015 (después del escándalo de las trampas) tenía como subtítulo «La superación del desempeño humano en la clasificación de ImageNet».[120] Casi al mismo tiempo, Microsoft anunció en un blog de investigación «un avance importante en la tecnología diseñada para identificar los objetos de una fotografía o un vídeo, que pone de relieve un sistema cuya precisión es equiparable y a veces superior a la de los seres humanos».[121] Aunque ambas empresas dejaron claro que hablaban de la precisión específicamente en ImageNet, los medios de comunicación tuvieron menos cautela y publicaron titulares sensacionalistas como «Los ordenadores ya reconocen y clasifican imágenes mejor que los humanos» y «Microsoft ha desarrollado un sistema informático que puede identificar objetos mejor que los humanos».[122]

Vamos a detenernos a analizar la afirmación de que las máquinas ya reconocen los objetos de ImageNet «mejor que los humanos». Esta frase se basa en la suposición de que los humanos tienen una tasa de error de aproximadamente el 5 por ciento, mientras que la tasa de error de las máquinas es (en el momento de escribir esto) de casi el 2 por ciento. ¿No confirma esto que las máquinas desempeñan esta tarea mejor que los humanos? Como suele ocurrir con los eslóganes más aireados sobre la IA, la frase está acompañada de unas cuantas reservas.

He aquí una de ellas. Cuando leemos que una máquina «identifica objetos correctamente», pensamos que, por ejemplo, ante una imagen de una pelota de baloncesto, el ordenador dice «pelota de baloncesto». Pero, claro, en ImageNet, «identificación correcta» solo quiere decir que entre las cinco mejores categorías aventuradas por el ordenador está la que es correcta. Si ante una imagen de una pelota de baloncesto el ordenador dice «bola de cróquet», «bikini», «jabalí», «pelota de baloncesto» y «furgoneta

en movimiento», en ese orden, la respuesta se considera correcta. No sé con qué frecuencia suceden este tipo de cosas, pero es llamativo que en la competición de ImageNet de 2017, el mejor caso de «precisión en primer lugar» —la mínima parte de imágenes de prueba en las que la categoría acertada ocupa el primer lugar de la lista— fue de aproximadamente el 82 por ciento, en comparación con el 98 por ciento de precisión de las cinco mejores. Que yo sepa, nadie ha comparado los aciertos de las máquinas y los humanos en una precisión de primer lugar.

He aquí otra reserva. Veamos esta frase: «Los humanos tienen una tasa de error de alrededor del 5 por ciento en ImageNet». Resulta que la palabra *humanos* no es del todo exacta, porque se refiere al resultado de un experimento en el que participó un solo ser humano, un tal Andrej Karpathy, que en aquel entonces era estudiante de posgrado en Stanford y estaba investigando sobre el aprendizaje profundo. Karpathy quería saber si podía entrenarse a sí mismo para competir contra las mejores ConvNet en ImageNet. Teniendo en cuenta que las ConvNet se entrenan con 1,2 millones de imágenes y luego se aplican a 150.000 imágenes de prueba, era una tarea abrumadora para un ser humano. Karpathy, que tiene un blog sobre IA muy popular, escribió sobre su experiencia:

> Acabé entrenándome con quinientas imágenes y luego estudié un conjunto de datos de prueba [reducido] de mil quinientas imágenes. El etiquetado [es decir, Karpathy adivinando cinco categorías por imagen] lo hice a un ritmo de aproximadamente una etiqueta por minuto, pero luego disminuyó con el tiempo. Solo disfruté de las primeras doscientas, más o menos, y continué con las demás en nombre de la ciencia… Algunas imágenes son fáciles de identificar, mientras que para otras con más matices (como las imágenes de razas de perros, pájaros o monos) hacen falta varios minutos de concentración. Aprendí a identificar muy bien razas de perros.[123]

Karpathy descubrió que se había equivocado aproximadamente en setenta y cinco de sus mil quinientas imágenes de prueba y se propuso analizar los errores cometidos; vio que se daban sobre todo en imágenes con varios objetos, en imágenes con razas concretas de perros, especies de pájaros o

plantas, etc., y en categorías de objetos que no se había dado cuenta de que estaban incluidas en las categorías de interés. Los tipos de errores que cometen las ConvNet son distintos: aunque también se confunden ante imágenes que contienen varios objetos, se diferencian de los humanos en que suelen pasar por alto los objetos pequeños, los que están distorsionados por filtros de color o de contraste que haya utilizado el fotógrafo y las «representaciones abstractas» de objetos, como un cuadro o una estatua de un perro o un perro de peluche. De modo que la afirmación de que los ordenadores han superado a los humanos en ImageNet hay que creérsela a medias.

Y aquí hay otra reserva que quizá les sorprenda. Cuando un humano dice que una foto contiene, por ejemplo, un perro, suponemos que es porque realmente ha visto un perro en la foto. Pero si una ConvNet dice acertadamente «perro», ¿cómo sabemos que está basando esa clasificación en el perro de la imagen? Quizá hay en la imagen alguna otra cosa —una pelota de tenis, un *frisbee*, un zapato mordisqueado— que ha encontrado asociada a los perros en las imágenes de entrenamiento y entonces la reconoce y da por supuesto que hay un perro en la foto. Muchas veces, este tipo de correlaciones ha acabado engañando a las máquinas.

Una cosa que podríamos hacer es pedir a la máquina que no solo asigne una categoría de objeto a una imagen, sino que también aprenda a dibujar un recuadro alrededor del objeto de interés, para que podamos saber que la máquina ha «visto» realmente el objeto. Eso es precisamente lo que la competición de ImageNet empezó a hacer en su segundo año con la «prueba de localización, que proporcionaba imágenes de entrenamiento con los recuadros ya dibujados (por trabajadores de Mechanical Turk) alrededor de los objetos de interés de cada imagen; en las imágenes de prueba, los programas competidores debían predecir cinco categorías de objetos, cada una con las coordenadas del recuadro correspondiente. Lo que tal vez es sorprendente es que, aunque las redes neuronales convolucionales

profundas funcionan muy bien en localización, su comportamiento sigue siendo considerablemente peor que en categorización, si bien las competiciones más recientes están centrándose precisamente en este problema.

Seguramente, las diferencias más importantes entre las ConvNet actuales y los seres humanos a la hora de reconocer objetos están en cómo aprenden y en lo sólido y fiable que resulta lo aprendido. Analizaré estas diferencias en el próximo capítulo.

Las reservas que he enumerado no pretenden quitar importancia a los asombrosos avances recientes de la visión por ordenador. No hay duda de que las redes neuronales convolucionales han tenido un éxito asombroso en este campo y otros, un éxito que no solo ha derivado en productos comerciales sino que también ha generado un verdadero sentimiento de optimismo en el mundillo de la IA. El propósito de mi análisis es ilustrar lo difícil que es la visión y aportar una nueva perspectiva a los avances logrados hasta ahora. Queda todavía mucho para que la inteligencia artificial «resuelva» el reconocimiento de objetos.

Más allá del reconocimiento de objetos

En este capítulo me he centrado en el reconocimiento de objetos porque es el área en la que la visión por ordenador ha avanzado más en los últimos tiempos. Pero es evidente que la visión consiste en mucho más que reconocer objetos. Si el objetivo de la visión por ordenador es «conseguir que una máquina describa lo que ve», entonces las máquinas tendrán que reconocer no solo los objetos, sino también las relaciones entre ellos y con el mundo. Si los «objetos» en cuestión son seres vivos, las máquinas tendrán que saber algo sobre sus acciones, sus objetivos, sus emociones, los próximos pasos más probables y todos los demás aspectos que componen la narración de una escena visual. Además, si verdaderamente queremos que las máquinas describan lo que ven, tendrán que utilizar el lenguaje. Los

investigadores sobre IA trabajan sin parar para conseguir que las máquinas hagan todo eso, pero, como de costumbre, las cosas «fáciles» son muy difíciles. Como declaró el experto en visión por ordenador Ali Farhadi a *The New York Times*, «todavía estamos muy muy lejos de la inteligencia visual, de entender las escenas y las acciones como lo hacen los seres humanos».[124]

¿Por qué estamos aún tan lejos de este objetivo? Parece que no es fácil separar la inteligencia visual del resto de la inteligencia, sobre todo del conocimiento general, la abstracción y el lenguaje, unas aptitudes en las que, curiosamente, intervienen partes del cerebro que tienen muchas conexiones de retroalimentación con la corteza visual. Además, es posible que los conocimientos necesarios para una inteligencia visual similar a la humana —por ejemplo, la capacidad de interpretar la foto de «la soldado y el perro» del principio del capítulo anterior— no se puedan adquirir de millones de imágenes descargadas de la web, sino que tengan que experimentarse de una u otra manera en el mundo real.

En el próximo capítulo voy a analizar con detalle el aprendizaje automático de la visión, sobre todo las diferencias entre la forma de aprender de los humanos y la de las máquinas, e intentaré averiguar qué han aprendido realmente las máquinas que hemos entrenado.

---

[107] De hecho, la retropropagación es un algoritmo que descubrieron de forma independiente varios grupos diferentes e, irónicamente, dada la función de la retropropagación como algoritmo de asignación de créditos, la atribución del mérito de su descubrimiento ha provocado una larga batalla entre los investigadores de redes neuronales.

[108] Citado en D. Hernandez, «Facebook's Quest to Build an Artificial Brain Depends on This Guy», *Wired*, 14 de agosto de 2014, www.wired.com/2014/08/deep-learningyann-lecun/.

[109] También hubo un concurso de «detección», en el que los programas además tenían que localizar objetos de las distintas categorías de las imágenes, y otros concursos especializados; aquí me centraré en el de clasificación.

[110] D. Gershgorn, «The Data That Transformed AI Research—and Possibly the World», *Quartz*, 26 de julio de 2017, qz.com/1034972/the-data-that-changed-thedirection-of-ai-research-and-possibly-the-world/.

[111] «About Amazon Mechanical Turk», www.mturk.com/help.

[112] L. Fei-Fei y J. Deng, «ImageNet: Where Have We Been? Where Are We Going?», diapositivas en image-net.org/challenges/talks_2017/imagenet_ilsvrc2017_v1.0.pdf.

[113] A. Krizhevsky, I. Sutskever y G. E. Hinton, «ImageNet Classification with Deep Convolutional Neural Networks», *Advances in Neural Information Processing Systems* 25 (2012), pp. 1097-1105.

[114] T. Simonite, «Teaching Machines to Understand Us», *Technology Review*, 5 de agosto de 2015, www.technologyreview.com/s/540001/teaching-machines-tounderstand-us/.

[115] Anuncio del concurso de reconocimiento visual a gran escala ImageNet, 2 de junio de 2015, www.image-net.org/challenges/LSVRC/announcement-June-2-2015.

[116] S. Chen, «Baidu Fires Scientist Responsible for Breaching Rules in High-Profile Supercomputer AI Test», *South China Morning Post*, edición internacional, 12 de junio de 2015, www.scmp.com/tech/science-research/article/1820649/chinas-baidufires-researcher-after-team-cheated-high-profile.

[117] Gershgorn, «Data That Transformed AI Research».

[118] Citado en Hernandez, «Facebook's Quest to Build an Artificial Brain Depends on This Guy».

[119] B. Agüera y Arcas, «Inside the Machine Mind: Latest Insights on Neuroscience and Computer Science from Google» (vídeo de la conferencia), Oxford Martin School, 10 de mayo de 2016, www.youtube.com/watch?v=v1dW7ViahEc.

[120] K. He *et al.*, «Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification», en *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1026-1034.

[121] A. Linn, «Microsoft Researchers Win ImageNet Computer Vision Challenge», *AI Blog*, Microsoft, 10 de diciembre de 2015, blogs.microsoft.com/ai/2015/12/10/microsoft-researchers-win-imagenet-computer-vision-challenge.

[122] A. Hern, «Computers Now Better than Humans at Recognising and Sorting Images», *Guardian*, 13 de mayo de 2015, https://www.theguardian.com/global/2015/may/13/baidu-minwa-supercomputer-better-than-humans-recognising-images; T. Benson, «Microsoft Has Developed a Computer System That Can Identify Objects Better than Humans», UPI, 14 de febrero de 2015, www.upi.com/ScienceNews/ 2015/02/14/Microsoft-has-developed-a-computer-system-that-can-identify-objectsbetter-than-humans/1171423959603.

[123] A. Karpathy, «What I Learned from Competing Against a ConvNet on ImageNet», 2 de septiembre de 2014, karpathy.github.io/2014/09/02/what-i-learned-fromcompeting-against-a-convnet-on-imagenet.

[124] S. Lohr, «A Lesson of Tesla Crashes? Computer Vision Can't Do It All Yet», *The New York Times*, 19 de septiembre de 2016.

06

Un análisis detallado de las máquinas que aprenden

El pionero del aprendizaje profundo Yann LeCun ha recibido muchos premios y elogios, pero quizá su máximo honor (aunque friki) sea que le dedicaran una cuenta paródica de Twitter, muy divertida y con muchos seguidores, que lleva el nombre de «Bored Yann LeCun» (Yann LeCun aburrido). La cuenta, que es anónima y se describe como «Reflexiones sobre el auge del aprendizaje profundo en el tiempo libre de Yann», suele terminar sus ingeniosos tuits con el *hashtag* #FeelTheLearn (Siente lo aprendido).[125]

De hecho, las noticias sobre los últimos avances de la IA en los medios de comunicación «sienten lo aprendido» cuando celebran el poder del aprendizaje profundo, con énfasis en el «aprendizaje». Nos dicen, por ejemplo, que «ahora podemos construir sistemas que aprenden a hacer tareas por sí solos»,[126] que «el aprendizaje profundo [permite] a los ordenadores enseñarse textualmente a sí mismos»[127] y que los sistemas de aprendizaje profundo aprenden «de manera similar al cerebro humano».[128]

En este capítulo examinaré con más detalle cómo aprenden las máquinas —en particular las ConvNet— y en qué se diferencian sus procesos de aprendizaje de los de los humanos. Además, analizaré en qué afectan las

diferencias entre el aprendizaje de las ConvNet y el de los humanos a la solidez y fiabilidad de lo aprendido.

Aprender por sí solas

El método de aprendizaje a partir de datos de las redes neuronales profundas ha demostrado tener más éxito, en general, que «la vieja estrategia de IA de toda la vida» en la que los programadores humanos elaboran unas reglas explícitas para obtener un comportamiento inteligente. Sin embargo, en contra de lo que se lee en algunos medios, el proceso de aprendizaje de las ConvNet se parece poco al de los humanos.

Como hemos visto, las mejores ConvNet aprenden mediante un procedimiento de aprendizaje supervisado: cambian gradualmente los pesos a medida que procesan los ejemplos del conjunto de datos de entrenamiento una y otra vez, a lo largo de muchas épocas (es decir, muchas repeticiones con los datos de entrenamiento), aprendiendo a clasificar cada entrada dentro de un conjunto fijo de posibles categorías de salida. Por el contrario, los niños, ya desde muy pequeños, aprenden un conjunto abierto de categorías y pueden reconocer casos de la mayoría de las categorías después de ver solo unos cuantos ejemplos. Además, los niños no aprenden de forma pasiva: hacen preguntas, piden información sobre las cosas que despiertan su curiosidad, deducen abstracciones y conexiones entre conceptos y, sobre todo, exploran el mundo.

No se puede decir que las ConvNet actuales aprenden «solas». Como vimos en el capítulo anterior, para que una ConvNet aprenda a hacer una tarea es necesario un enorme esfuerzo humano que permita recopilar, organizar y etiquetar los datos, además de diseñar todos los aspectos de su arquitectura. Aunque las ConvNet utilizan la retropropagación para aprender sus «parámetros» (es decir, los pesos) a partir de los ejemplos de entrenamiento, el aprendizaje es posible gracias a una serie de «hiperparámetros», un término genérico que abarca todos los aspectos de la

red que el ser humano debe configurar solo para que el aprendizaje pueda comenzar. Entre esos hiperparámetros están el número de capas de la red, el tamaño de los «campos receptivos» de las unidades en cada capa, cuánto debe cambiar cada peso durante el aprendizaje (la llamada tasa de aprendizaje) y muchos otros detalles técnicos del proceso de entrenamiento. Esta parte de la configuración de una ConvNet se denomina ajuste de los hiperparámetros. Hay muchos valores que ajustar y complicadas decisiones de diseño que tomar, y esos ajustes y diseños tienen una relación compleja que influye en el comportamiento final de la red. Además, las decisiones sobre esos ajustes y diseños deben volver a tomarse ante cada tarea para la que es entrenada una red.

Ajustar los hiperparámetros puede parecer bastante rutinario, pero es absolutamente crucial hacerlo bien para el éxito de las ConvNet y otros sistemas de aprendizaje automático. Como el diseño de estas redes no está cerrado, en general no es posible establecer de forma automática todos los parámetros y diseños, ni siquiera con la búsqueda automatizada. Muchas veces hace falta una especie de conocimiento cabalístico que los estudiantes de aprendizaje automático adquieren tanto a través de su formación con expertos como de la experiencia adquirida con tanto esfuerzo. Como dice Eric Horvitz, director del laboratorio de investigación de Microsoft: «Ahora mismo, lo que estamos haciendo no es una ciencia, sino una especie de alquimia».[129] Y estos «encantadores de redes» forman un club pequeño y selecto: según Demis Hassabis, cofundador de Google DeepMind, «sacar lo mejor de estos sistemas es casi un arte… No hay más que unos cientos de personas en el mundo capaces de hacerlo realmente bien».[130]

En realidad, el número de expertos en aprendizaje profundo está aumentando a toda velocidad; muchas universidades ofrecen ya cursos sobre el tema, y hay cada vez más empresas con sus propios programas de formación en aprendizaje profundo para sus empleados. Pertenecer al club del aprendizaje profundo puede ser bastante lucrativo. En una conferencia a

la que asistí hace poco, un directivo del grupo de productos de IA de Microsoft habló sobre la campaña de la empresa para contratar a jóvenes ingenieros especializados en aprendizaje profundo: «Si un chico sabe entrenar cinco capas de redes neuronales, puede pedir un salario de cinco cifras. Si sabe entrenar cincuenta capas, puede pedir un salario de siete cifras».[131] Por suerte para ese chico al que le espera tanta riqueza, las redes todavía no pueden aprender por sí solas.

Macrodatos

No es ningún secreto que el aprendizaje profundo necesita grandes volúmenes de datos. «Grandes» quiere decir más de un millón de imágenes de entrenamiento etiquetadas en ImageNet. ¿De dónde proceden todos esos datos? La respuesta es, por supuesto, que de ti y probablemente de todos tus conocidos. Las aplicaciones modernas de visión por ordenador solo son posibles gracias a los miles de millones de imágenes que los usuarios de internet suben y (a veces) etiquetan con un texto que identifica lo que aparece. ¿Alguna vez han subido una foto de un amigo a Facebook y la han comentado? Facebook se lo agradece. Esa imagen y ese texto pueden haber servido para entrenar su sistema de reconocimiento facial. ¿Alguna vez han subido una imagen a Flickr? En ese caso, es posible que su imagen forme parte del conjunto de entrenamiento de ImageNet. ¿Alguna vez han identificado una imagen para demostrar en una web que no son un robot? Esa identificación quizá ha ayudado a Google a etiquetar una imagen para usarla en el entrenamiento de su sistema de búsqueda de imágenes.

Las grandes empresas tecnológicas ofrecen muchos servicios gratuitos en el ordenador y el teléfono móvil: búsqueda en internet, videollamadas, correo electrónico, redes sociales, asistentes personales automatizados…, una lista interminable. ¿Qué salen ganando? Quizá han oído decir que su verdadero producto son sus usuarios (como usted y como yo); los clientes son los anunciantes que captan nuestra atención y adquieren información

sobre nosotros mientras utilizamos estos servicios «gratuitos». Pero hay una segunda respuesta: cuando utilizamos los servicios de empresas tecnológicas como Google, Amazon y Facebook, estamos proporcionando directamente a esas empresas ejemplos —imágenes, vídeos, mensajes de texto o voz— que pueden aprovechar para entrenar mejor sus programas de IA. Y esos programas mejorados atraen a más usuarios (y, por tanto, recogen más datos), lo que hace que los anunciantes puedan dirigir sus anuncios de forma más eficaz. Además, los ejemplos de entrenamiento que les proporcionamos pueden servir para entrenar y ofrecer a otras empresas servicios «de oficina», como la visión por ordenador y el procesamiento del lenguaje natural, a cambio de dinero.

Se ha escrito mucho sobre la ética de estas grandes empresas que utilizan los datos que creamos nosotros (por ejemplo, todas las imágenes, los vídeos y los textos que colgamos en Facebook) para entrenar programas y vender productos sin decírnoslo ni compensarnos. Es un debate importante, pero se sale del ámbito de este libro.[132] Lo que me interesa aquí es que la dependencia de extensas colecciones de datos de entrenamiento etiquetados es una diferencia más entre el aprendizaje profundo y el aprendizaje humano.

Con la proliferación de sistemas de aprendizaje profundo en aplicaciones del mundo cotidiano, las empresas necesitan nuevos conjuntos de datos etiquetados para entrenar redes neuronales profundas. Un ejemplo destacable son los vehículos autónomos. Estos coches necesitan una visión por ordenador avanzada para reconocer los carriles de la carretera, los semáforos, las señales de *stop* y otros elementos, así como para distinguir y seguir la pista de distintos tipos de posibles obstáculos: otros coches, peatones, ciclistas, animales, conos de tráfico, cubos de basura volcados, matojos rodadores y cualquier otra cosa con la que no conviene chocar. Los coches autónomos tienen que aprender a identificar esos objetos —con sol, lluvia, nieve o niebla, de día o de noche— y a determinar cuáles pueden

moverse y cuáles no. El aprendizaje profundo facilita esa tarea, al menos en parte, pero, como en otros ámbitos, necesita una enorme cantidad de ejemplos de entrenamiento.

Las empresas de vehículos autónomos recogen esos ejemplos de entrenamiento en un sinnúmero de horas de vídeo grabadas por cámaras desde coches que circulan en medio del tráfico de calles y carreteras. Los coches pueden ser prototipos de conducción autónoma que las empresas están probando o, en el caso de Tesla, coches conducidos por clientes que, al comprar un vehículo, tienen que aceptar una política de intercambio de datos con la empresa.[133]

Los propietarios de Tesla no tienen obligación de etiquetar todos los objetos que aparecen en los vídeos grabados por sus coches. Pero alguien tiene que hacerlo. En 2017, el *Financial Times* informó de que «la mayoría de las empresas que desarrollan esta tecnología emplean a cientos e incluso miles de personas, muchas veces en centros deslocalizados en India o China, cuyo trabajo consiste en enseñar a los coches robot a reconocer peatones, ciclistas y otros obstáculos. Los empleados marcan o "etiquetan" manualmente miles de horas de vídeo, a menudo fotograma a fotograma».[134] Han nacido nuevas empresas que proporcionan el servicio del etiquetado de datos; por ejemplo, Mighty AI ofrece «los datos etiquetados que necesitas para entrenar tus modelos de visión por ordenador» y promete «anotadores conocidos, verificados y de confianza, especializados en datos de conducción autónoma».[135]

La cola larga

El método de aprendizaje supervisado, que utiliza grandes conjuntos de datos y ejércitos de anotadores humanos, funciona bien al menos para parte de las aptitudes visuales que necesitan los coches autónomos (muchas empresas están investigando también el uso de programas de conducción simulada, similares a los videojuegos, para reforzar el entrenamiento). Pero

¿qué sucede en otros aspectos de la vida? Prácticamente todos los que trabajan en el campo de la IA coinciden en que el aprendizaje supervisado no es un método viable para la IA de espectro general. Como ha advertido el prestigioso investigador sobre IA Andrew Ng: «La necesidad de tantos datos es una de las principales limitaciones actuales [del aprendizaje profundo]».[136] Yoshua Bengio, otro destacado investigador de IA, está de acuerdo: «No es realista pensar que podemos etiquetar todo lo que hay en el mundo y explicar meticulosamente hasta el último detalle al ordenador».[137]

El problema se ve agravado por la llamada cuestión de las colas largas: la gran variedad de posibles situaciones inesperadas con las que puede encontrarse un sistema de IA. La figura 13 (en la página siguiente) ilustra este fenómeno mostrando la probabilidad de que se produzcan varias situaciones hipotéticas con las que puede encontrarse un coche autónomo, por ejemplo, circulando durante un día. Las situaciones muy corrientes, como toparse con un semáforo en rojo o una señal de *stop*, se clasifican como muy probables; las situaciones con una probabilidad media son, por ejemplo, cristales rotos y bolsas de plástico azotadas por el viento, que no se encuentran todos los días (dependiendo de por dónde circule el coche), pero no son infrecuentes. Menos probable es que el coche autónomo se encuentre con una carretera inundada o con los carriles tapados por la nieve, y todavía menos que se tope con un muñeco de nieve en medio de una autopista.

Se me han ocurrido estas situaciones y he calculado sus probabilidades; seguro que a cada persona se le ocurren muchas más. Probablemente, cada uno de estos coches es seguro: al fin y al cabo, en total, los coches autónomos experimentales han recorrido ya millones de kilómetros y han causado un número relativamente pequeño de accidentes (aunque algunos han sido mortales y han tenido gran repercusión). Ahora bien, cuando los coches autónomos se generalicen, aunque cada situación concreta

improbable sea, por definición, muy improbable, hay tantas situaciones posibles en el mundo de la conducción y tantos coches que sí es probable que algún coche autónomo, en algún lugar y en algún momento, se encuentre con una de esas situaciones.

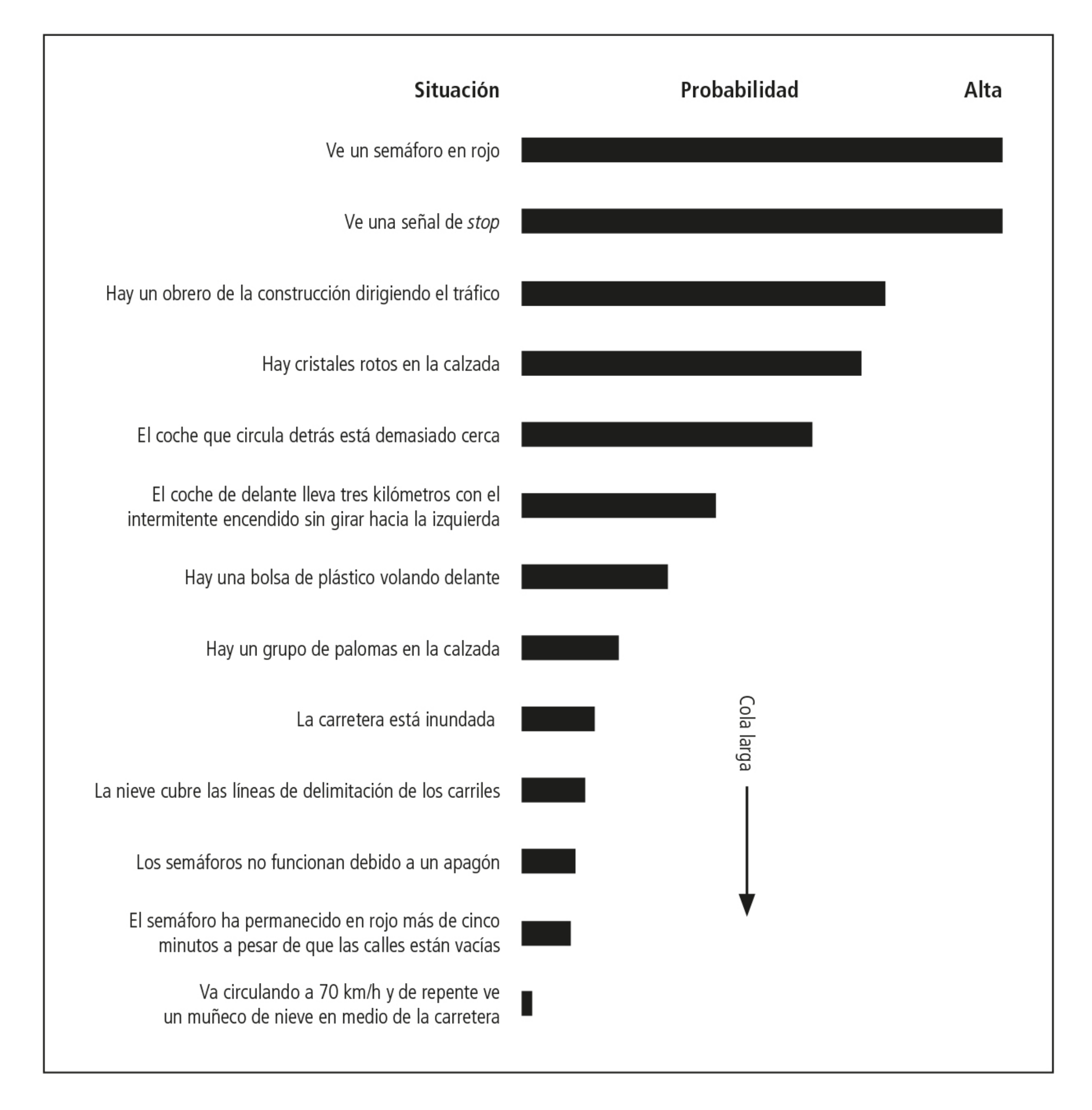


Figura 13. Posibles situaciones con las que puede encontrarse un coche autónomo, clasificadas por probabilidad, lo que ilustra la «cola larga» de situaciones improbables.

El término *cola larga* procede de la estadística, en la que ciertas distribuciones de probabilidad tienen una forma similar a la de la figura 13:

la larga lista de situaciones muy improbables (pero posibles) es la «cola» de la distribución. (Las situaciones que forman la cola se denominan a veces casos extremos). La mayoría de los ámbitos reales en los que actúa la IA contienen este fenómeno de cola larga: los acontecimientos del mundo real suelen ser predecibles, pero queda una larga cola de sucesos inesperados y poco probables. Eso supone un problema si, para proporcionar a nuestro sistema de IA su conocimiento del mundo, nos fiamos únicamente del aprendizaje supervisado; las situaciones de la cola no aparecen suficientes veces en los datos de entrenamiento, si es que aparecen, por lo que hay más probabilidades de que el sistema cometa errores cuando se encuentre con esos casos inesperados.



Figura 14. Hubo informaciones de que las líneas de sal en una autopista, antes de una tormenta de nieve prevista, confundían a la función Autopilot de Tesla.

Mostraré dos ejemplos reales. En marzo de 2016 se preveía una gran tormenta de nieve en el nordeste de Estados Unidos y en Twitter aparecieron informaciones de que el modo Autopilot de los vehículos Tesla, que permite una conducción autónoma limitada, confundía las líneas de los carriles y los montones de sal colocados en línea en la autopista en previsión de la tormenta (figura 14). En febrero de 2016, uno de los prototipos de coches autónomos de Google, que estaba girando a la derecha, tuvo que virar a la izquierda para evitar unos sacos de arena en el arcén derecho de una carretera de California y golpeó con la parte delantera izquierda un autobús público que circulaba por el carril izquierdo. Cada vehículo había contado con que el otro le cediera el paso (quizá el conductor del autobús pensaba que un conductor humano se sentiría intimidado por el autobús, mucho más grande).

Las empresas que desarrollan la tecnología de vehículos autónomos son muy conscientes del problema de la cola larga: sus equipos no paran de imaginar posibles situaciones de cola larga y crean sin cesar nuevos ejemplos de formación y estrategias codificadas especialmente para todas las situaciones poco probables que se les ocurren. Pero está claro que es imposible entrenar o codificar un sistema para todas las situaciones posibles.

Una solución que suele proponerse es que los sistemas de IA utilicen el aprendizaje supervisado con pequeñas cantidades de datos etiquetados y adquieran todo lo demás mediante aprendizaje no supervisado. «Aprendizaje no supervisado» engloba un vago conjunto de métodos para aprender categorías o acciones sin datos etiquetados, como los métodos para agrupar ejemplos con arreglo a su similitud o para aprender una nueva categoría por analogía con categorías conocidas, entre otros. Como explicaré en un capítulo posterior, a los humanos se les da muy bien

percibir similitudes y analogías abstractas, pero hasta ahora no existen métodos que hayan tenido mucho éxito en este tipo de aprendizaje no supervisado de la IA. El propio Yann LeCun reconoce que «el aprendizaje no supervisado es la materia oscura de la IA». En otras palabras, para la IA general, casi todo el aprendizaje tendrá que ser no supervisado, pero nadie ha dado todavía con el tipo de algoritmos necesarios para hacer ese aprendizaje no supervisado con buenos resultados.

Los humanos cometen errores constantemente, incluso (o especialmente) al volante; cualquiera de nosotros podría haber chocado con ese autobús si hubiéramos tenido que sortear los sacos de arena. Pero los humanos también tienen una competencia fundamental de la que carecen todos los sistemas de IA actuales: el sentido común. Tenemos un amplio conocimiento de fondo del mundo, tanto en el aspecto físico como en el social. Tenemos una idea bastante clara de cómo es probable que vayan a comportarse los objetos —tanto inanimados como vivos—, y utilizamos ese conocimiento para decidir cómo actuar en una situación determinada. Podemos deducir el motivo de los montones de sal en la carretera aunque nunca hayamos conducido con nieve. Sabemos relacionarnos socialmente con otros seres humanos, así que podemos hacer contacto visual, señales con las manos y otros gestos para compensar un semáforo estropeado durante un apagón. En general, sabemos que debemos ceder el paso a un autobús de transporte público, aunque en teoría tengamos prioridad. He puesto un ejemplo del tráfico, pero los seres humanos utilizamos el sentido común —casi siempre de forma subconsciente— en todas las facetas de la vida. Mucha gente cree que hasta que los sistemas de IA no tengan el mismo sentido común que los humanos, no podremos confiar en que sean totalmente autónomos en situaciones complejas del mundo real.

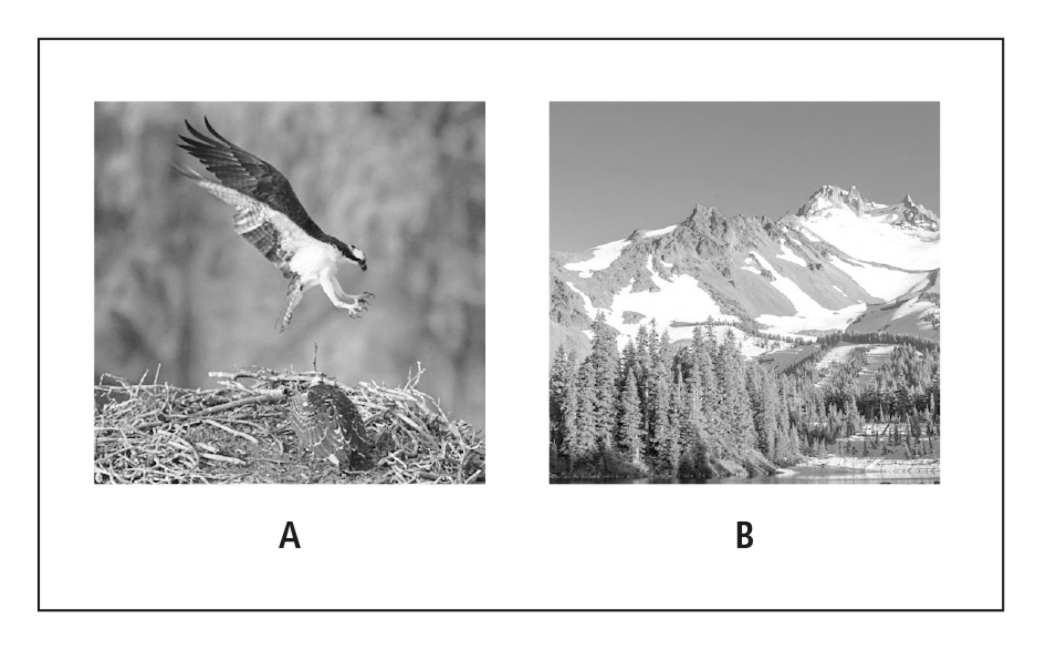


Figura 15. Ilustración de la tarea de clasificación como «animal» y «no animal». Obsérvese el fondo borroso en la imagen de la izquierda.

¿Qué ha aprendido mi red?

Hace unos años, Will Landecker, entonces estudiante de posgrado en mi grupo de investigación, entrenó una red neuronal profunda para clasificar fotografías en dos categorías: «contiene un animal» y «no contiene un animal». La red se entrenó con fotos como las de la figura 15 y obtuvo muy buenos resultados con las imágenes de prueba. Pero ¿qué aprendió realmente la red? Will llevó a cabo un análisis minucioso y se encontró con una respuesta inesperada: en parte, la red había aprendido a clasificar las imágenes con fondo borroso como «contiene un animal», tanto si había verdaderamente un animal como si no.[138] Las fotos de naturaleza de los conjuntos de entrenamiento y de prueba seguían una regla importante en fotografía: el foco es el sujeto de la foto. Cuando el sujeto de la foto es un animal, el animal es el foco y el fondo está borroso, como en la figura 15A. Cuando el sujeto de la foto es el fondo, como en la figura 15B, no se desenfoca nada. Para desilusión de Will, su red no había aprendido a reconocer animales, sino que utilizaba pistas más simples —como los

fondos borrosos— que estadísticamente estaban asociadas a la presencia de animales.

Este es un ejemplo de un fenómeno habitual en el aprendizaje automático. La máquina aprende lo que observa en los datos, no lo que nosotros (los humanos) podríamos observar. Si hay asociaciones estadísticas en los datos de entrenamiento, aunque sean irrelevantes para la tarea en cuestión, la máquina aprenderá eso, no lo que nosotros queríamos que aprendiera. Si la máquina hace una prueba con nuevos datos que incluyan las mismas asociaciones estadísticas, parecerá que ha aprendido a resolver la tarea. Pero la máquina puede fallar de forma inesperada, como le ocurrió a la red de Will con las imágenes de animales que no tenían un fondo borroso. En el lenguaje del aprendizaje automático, la red de Will se «sobreajustó» a su conjunto de entrenamiento específico y, por tanto, no pudo aplicar bien lo aprendido a otras imágenes que no fueran las del entrenamiento.

En los últimos años, varios equipos han investigado si las ConvNet entrenadas en ImageNet y otros grandes conjuntos de datos se han sobreajustado de esa forma a sus datos de entrenamiento. Un grupo ha demostrado que si las ConvNet se entrenan con imágenes descargadas de internet (como las de ImageNet), tienen más problemas con imágenes tomadas por un robot mientras se desplaza por una casa con una cámara.[139] Parece que las vistas aleatorias de objetos domésticos pueden ser muy distintas de las fotos que la gente cuelga en la web. Otros grupos han demostrado que una modificación superficial de las fotos, como difuminar o llenar de puntos una imagen, cambiar algunos colores o rotar varios objetos de la escena, pueden hacer que las ConvNet cometan errores significativos cuando esas perturbaciones no impiden que los humanos reconozcan los objetos.[140] Esta inesperada fragilidad de las ConvNet —incluso de aquellas que supuestamente «superan a los humanos en el reconocimiento

de objetos»— indica que se están ajustando en exceso a sus datos de entrenamiento y aprendiendo algo distinto de lo que intentamos enseñarles.

Una IA sesgada

La poca fiabilidad de las ConvNet puede desembocar en errores embarazosos y quizá perjudiciales. En 2015, Google vivió una situación de pesadilla para su reputación cuando presentó una función de etiquetado automático de fotos (mediante una ConvNet) en su aplicación Fotos. Además de etiquetar correctamente imágenes con descripciones genéricas como «aviones», «coches» y «graduación», la red neuronal asignó a un selfi en el que aparecían dos afroamericanos la etiqueta de «gorilas», como se muestra en la figura 16 (en la página siguiente). (Después de pedir muchas disculpas, la solución inmediata de la empresa fue eliminar la etiqueta «gorilas» de la lista de categorías posibles).

Estos errores de clasificación, repugnantes y muy ridiculizados, son embarazosos para las empresas implicadas, pero con frecuencia se han visto errores más sutiles debidos a sesgos raciales o de género en sistemas de visión basados en el aprendizaje profundo. Los sistemas comerciales de reconocimiento facial, por ejemplo, tienden a ser más precisos con los rostros masculinos blancos que con los rostros femeninos o no blancos.[141] Los programas de detección facial tienden a veces a pasar por alto los rostros de piel oscura y a clasificar los rostros asiáticos como «parpadeantes» (figura 17).



Figura 16. Etiquetas asignadas a fotos por el etiquetador automático de fotos de Google, incluida la infame etiqueta de «gorilas».

Kate Crawford, investigadora de Microsoft y activista en favor de la equidad y la transparencia en la IA, destaca que los rostros contenidos en un conjunto de datos muy utilizado para entrenar sistemas de reconocimiento facial son en un 77,5 por ciento de hombres y en un 83,5 por ciento de blancos. Esto no es nada raro, porque las imágenes se descargaron a partir de búsquedas en internet, donde existe un sesgo a favor de personas famosas o poderosas, que son predominantemente blancas y masculinas.

Por supuesto, estos sesgos en los datos de entrenamiento de la IA reflejan los sesgos de nuestra sociedad, pero la generalización en el mundo real de sistemas de IA entrenados con datos sesgados puede agravarlos y causar daños considerables. Por ejemplo, los sistemas de reconocimiento facial se utilizan cada vez más como forma «segura» de identificar a las personas en las transacciones con tarjetas de crédito, los controles de los aeropuertos y las cámaras de seguridad, y puede que no falte mucho para que se utilicen

como método de identificación en los sistemas de votación, entre otras aplicaciones. La más mínima diferencia de precisión entre unos grupos raciales y otros puede tener consecuencias perjudiciales para los derechos civiles y el acceso a servicios vitales.



Figura 17. Ejemplo de un programa de detección de rostros por cámara que identifica un rostro asiático como «parpadeante».

En conjuntos de datos específicos es posible mitigar estos sesgos si se encarga a seres humanos que se aseguren de que las fotos (o cualquier otro tipo de datos) mantengan el equilibrio en su representación de, por ejemplo, grupos raciales o de género. Pero para ello es necesario que las personas que organizan los datos sean conscientes de ello y realicen su tarea de forma cuidadosa. Además, muchas veces es difícil detectar los sesgos sutiles y sus efectos. Por ejemplo, un grupo de investigación observó que su sistema de IA —entrenado con un gran conjunto de fotos de personas en diferentes situaciones— a veces se equivocaba y clasificaba a un hombre como «mujer» cuando aparecía en una cocina, un entorno en el que el conjunto de datos tenía más ejemplos de mujeres.[142] En general, este tipo de sesgo sutil puede ser evidente *a posteriori*, pero es difícil de detectar con antelación.

El problema del sesgo en las aplicaciones de IA ha sido objeto de mucha atención en los últimos tiempos, con numerosos artículos, talleres e incluso institutos de investigación académica dedicados a este tema. ¿Los conjuntos de datos que se utilizan para entrenar la IA deben reflejar fielmente los sesgos de nuestra sociedad —como suelen hacer en la actualidad—, o habría que retocarlos específicamente para cumplir objetivos de reforma social? ¿Y quién debería poder concretar los objetivos o hacer los retoques?

Enseñar cómo se ha hecho

¿Recuerdan cuando, en el colegio, el profesor escribía en rojo «enséñame cómo lo has hecho» en los deberes de matemáticas? Para mí, explicar cómo lo había hecho era la parte menos divertida de aprender matemáticas, pero seguramente era la más importante, porque decir cómo había deducido mi respuesta demostraba que verdaderamente había entendido lo que estaba haciendo, que había captado las abstracciones correctas y había llegado a la respuesta como era debido. Además, enseñar cómo lo había hecho también ayudaba a mi profesor a saber por qué cometía determinados errores.

En general, se puede confiar en que una persona sabe lo que hace si es capaz de explicar cómo ha llegado a una respuesta o a una decisión. Sin embargo, «enseñar cómo lo han hecho» no es algo que las redes neuronales profundas —la base de los sistemas modernos de IA— puedan hacer así como así. Volvamos a la tarea de identificar objetos como «perros» y «gatos» que describí en el capítulo 4. Recordemos que una red neuronal convolucional decide qué objeto hay en una imagen que le llega mediante una secuencia de operaciones matemáticas (convoluciones) propagadas a través de muchas capas. En una red de tamaño razonable, pueden hacerse hasta miles de millones de operaciones aritméticas. Sería fácil programar el ordenador para que imprima una lista de todas las sumas y multiplicaciones hechas por una red para una entrada determinada, pero esa lista no nos permitiría saber absolutamente nada de cómo ha llegado la red a su

respuesta. Una lista de mil millones de operaciones no es una explicación que un humano pueda entender. Ni siquiera los humanos que entrenan redes profundas, en general, pueden mirar bajo el capó y explicar las decisiones que toman sus redes. La revista *Technology Review*, del MIT, llamó a esta impenetrabilidad «el oscuro secreto en el corazón de la IA».[143] Lo que preocupa es que si no entendemos cómo funcionan los sistemas de IA, no podemos confiar realmente en ellos ni predecir en qué circunstancias cometerán errores.

Los seres humanos tampoco pueden explicar siempre sus procesos mentales, y, en general, no es posible mirar «bajo el capó» y hurgar en el cerebro de otra persona (o en sus «instintos») para averiguar cómo ha llegado a una decisión concreta. Pero los humanos tendemos a confiar en que otros humanos dominan tareas cognitivas básicas como el reconocimiento de objetos y la comprensión del lenguaje. En parte, confiamos en los demás cuando creemos que su forma de pensar es como la nuestra. En la mayoría de los casos, suponemos que los demás seres humanos con los que nos encontramos han tenido experiencias vitales bastante similares a las nuestras y, por tanto, que se basan en los mismos conocimientos básicos, creencias y valores que nosotros a la hora de percibir, describir y tomar decisiones sobre el mundo. En resumen, en nuestra relación con otras personas, tenemos lo que los psicólogos llaman una teoría de la mente: un modelo de los conocimientos y objetivos de la otra persona en situaciones concretas. Nadie tiene una «teoría de la mente» similar en relación con sistemas de IA como las redes profundas, por lo que es más difícil confiar en ellos.

No es extraño que uno de los nuevos campos más de moda de la IA sea el que llaman «IA explicable», «IA transparente» o «aprendizaje automático interpretable». Estos términos designan la investigación sobre cómo conseguir que los sistemas de IA —en especial las redes profundas— expliquen sus decisiones de manera comprensible para los humanos. Los

investigadores de este terreno han concebido astutas formas para visualizar los elementos que ha aprendido una red neuronal convolucional y, en algunos casos, determinar qué partes de la información de entrada pesan más en la decisión de salida. La IA explicable es un campo que avanza con rapidez, pero todavía no se ha conseguido crear un sistema de aprendizaje profundo capaz de explicarse a sí mismo en términos humanos.

Engañar a las redes neuronales profundas

Hay otra dimensión más en la cuestión de la fiabilidad de la IA: los investigadores han descubierto que para los humanos es asombrosamente fácil engañar a las redes neuronales profundas para que cometan errores. Es decir, si queremos engañar deliberadamente a un sistema de este tipo, resulta que hay una terrible cantidad de maneras de hacerlo.

Engañar a los sistemas de inteligencia artificial no es nuevo. Quienes llenan de *spam* nuestros correos electrónicos, por ejemplo, llevan décadas en una carrera armamentística con los programas centrados en su detección. Pero los ataques a los que parecen ser vulnerables los sistemas de aprendizaje profundo son al mismo tiempo más sutiles y más preocupantes.

¿Recuerdan AlexNet, de la que hablé en el capítulo 5? Era la red neuronal convolucional que ganó el concurso de ImageNet de 2012 e inició el dominio de las ConvNet en gran parte del mundo de la IA actual. Recordarán que la precisión de AlexNet (con las cinco mejores conjeturas) en ImageNet fue del 85 por ciento, con lo que eliminó a todos los demás competidores y asombró al mundo de la visión por ordenador. Sin embargo, un año después de la victoria de AlexNet, apareció un artículo de investigación escrito por Christian Szegedy, de Google, y varios otros, con el título engañosamente suave de «Intrigantes propiedades de las redes neuronales».[144] Una de las «propiedades intrigantes» descritas en el ensayo era que resultaba fácil engañar a AlexNet.

En concreto, los autores del artículo habían descubierto que podían coger una foto de ImageNet que AlexNet había clasificado acertadamente y con gran seguridad (por ejemplo, «autobús escolar») y distorsionarla con cambios muy pequeños y específicos en sus píxeles, de modo que la imagen distorsionada les pareciera completamente igual a los humanos, pero AlexNet ahora la clasificara con un grado de seguridad muy alto como algo completamente diferente (por ejemplo, «avestruz»). A la imagen distorsionada le dieron el nombre de «ejemplo antagónico». La figura 18 muestra varios ejemplos de imágenes originales y sus gemelas antagónicas. ¿No notan la diferencia? ¡Enhorabuena! Se ve que son humanos.



Figura 18. Ejemplos originales y «antagónicos» para AlexNet. La imagen de la izquierda de cada par muestra la imagen original, correctamente clasificada por AlexNet. La imagen de la derecha de cada par muestra el ejemplo antagónico derivado de esa imagen (se han hecho pequeñas modificaciones en los píxeles, pero a los humanos la nueva imagen les parece idéntica a la original).

Szegedy y sus colaboradores crearon un programa informático que, con cualquier foto de ImageNet correctamente clasificada por AlexNet, podía encontrar cambios específicos en la foto para crear un nuevo ejemplo

antagónico que a los humanos les pareciera inalterado pero que hiciera que AlexNet asignara una categoría incorrecta con la máxima seguridad.

Es importante señalar que Szegedy y sus colaboradores vieron que esta vulnerabilidad a los ejemplos antagónicos no era exclusiva de AlexNet, demostrando que otras ConvNet —con diferentes arquitecturas, hiperparámetros y conjuntos de entrenamiento— presentaban vulnerabilidades similares. Llamar a esto una «propiedad intrigante» de las redes neuronales es más o menos como decir que un agujero en el casco de un crucero de lujo es una «faceta del barco que da que pensar». Intrigante, sí, y hace falta investigar más, pero si no se arregla la fuga, el barco se va a pique.

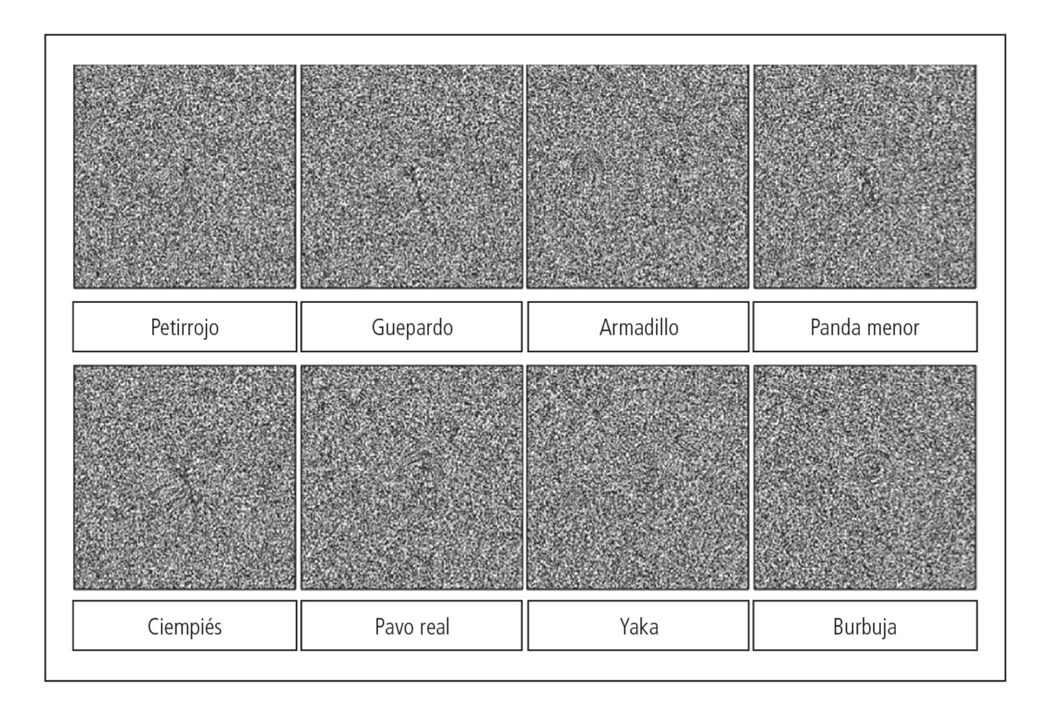


Figura 19. Ejemplos de imágenes creadas por un algoritmo genético específicamente para engañar a una red neuronal convolucional (ConvNet). En cada caso, AlexNet (entrenada con el conjunto de datos de entrenamiento de ImageNet) asignó una seguridad superior al 99 por ciento a la respuesta de que la imagen era un caso de la categoría mostrada.

Poco después de la publicación del artículo de Szegedy y sus colegas, un grupo de la Universidad de Wyoming publicó un artículo con un título más

directo: «Las redes neuronales profundas son fáciles de engañar».[145] Utilizando un método computacional inspirado en la biología denominado algoritmos genéticos,[146] el grupo de Wyoming fue capaz de «desarrollar» por ordenador imágenes que a los humanos les parecían ruido blanco pero a las que AlexNet y otras redes neuronales convolucionales asignaban categorías concretas de objetos con una seguridad superior al 99 por ciento. La figura 19 muestra algunos ejemplos. El grupo de Wyoming observó que las redes neuronales profundas (DNN, por sus siglas en inglés) «ven estos objetos como ejemplos casi perfectos de imágenes reconocibles», lo que «[suscita] dudas sobre la verdadera capacidad de generalización de las DNN y las posibilidades de que se haga un uso de las soluciones que emplean DNN que acabe saliendo caro [es decir, aplicaciones maliciosas]».[147]

De hecho, estos dos artículos y otros descubrimientos posteriores en este sentido suscitaron no solo dudas sino auténtica alarma en el mundo del aprendizaje profundo. Si los sistemas de aprendizaje profundo, tan eficaces en visión por ordenador y otras tareas, pueden ser engañados tan fácilmente con manipulaciones que no confunden a los humanos, ¿cómo podemos decir que estas redes «aprenden como los humanos» o «igualan o superan a los humanos» en sus capacidades? Está claro que aquí estamos ante algo muy distinto de la percepción humana. Y si estas redes se van a utilizar para la visión por ordenador en el mundo real, más vale que nos aseguremos de que están protegidas contra los piratas informáticos que utilizan este tipo de manipulaciones para engañarlas.

Todo esto ha revitalizado la pequeña comunidad investigadora que se dedica al «aprendizaje antagónico», es decir, al desarrollo de estrategias de defensa contra posibles antagonistas (humanos) que podrían atacar los sistemas de aprendizaje automático. Los investigadores sobre aprendizaje antagónico suelen empezar por mostrar formas posibles de atacar los sistemas actuales, y algunas de las demostraciones recientes han sido

asombrosas. En el campo de la visión por ordenador, un grupo de investigadores ha desarrollado un programa capaz de crear monturas de gafas con dibujos específicos que engañan a un sistema de reconocimiento facial para que se equivoque e identifique al usuario como otra persona (figura 20, en la página siguiente).[148] Otro grupo ha desarrollado unas pegatinas pequeñas y discretas que pueden colocarse en una señal de tráfico y hacen que un sistema de visión basado en ConvNet —del tipo de los utilizados en los coches autónomos— clasifique erróneamente la señal (por ejemplo, identifica una señal de *stop* como una señal de límite de velocidad).[149] Un tercer grupo ha presentado un posible ataque antagónico contra redes neuronales profundas empleadas en el análisis de imágenes médicas: demostraron que no es difícil alterar una imagen de rayos X o de microscopio de forma imperceptible para los humanos pero que hace que una red cambie su dictamen de, por ejemplo, un 99 por ciento de seguridad en que la imagen no muestra cáncer a un 99 por ciento de seguridad en que sí hay cáncer.[150] Este grupo subraya que el personal hospitalario u otros profesionales podrían utilizar ese tipo de ataques para crear diagnósticos fraudulentos y así cobrar a las compañías de seguros por más (y lucrativas) pruebas de diagnóstico.




Figura 20. Un investigador de IA (izquierda) lleva monturas de gafas con un patrón especialmente diseñado para que una red neuronal profunda de reconocimiento facial,

entrenada con rostros de famosos, clasifique con seguridad la foto de la izquierda como la actriz Milla Jovovich (derecha). El artículo en el que se describe este estudio da muchos otros ejemplos de suplantación de identidad utilizando patrones de montura de gafas «antagónicos».

Estos son solo algunos ejemplos de los posibles ataques que han imaginado diversos grupos de investigación. Muchos de ellos exhiben una solidez asombrosa: funcionan en varias redes distintas, incluso cuando se las entrena con conjuntos de datos diferentes. Y la visión por ordenador no es el único campo en el que se puede engañar a las redes; los investigadores también han diseñado ataques que engañan a las redes neuronales profundas relacionadas con el lenguaje en aspectos como el reconocimiento del habla y el análisis de texto. Es de suponer que a medida que estos sistemas se vayan extendiendo en el mundo real, los usuarios malintencionados descubran en ellos muchas otras vulnerabilidades.

Aprender a comprender estos posibles ataques y defenderse de ellos es un área actual de investigación importante, pero, aunque se han encontrado soluciones para tipos concretos de ataques, todavía no existe un método de defensa general. Como en cualquier otro campo de la seguridad informática, los avances conseguidos hasta ahora son más bien como un «juego del topo», en el que se detecta y se defiende un agujero de seguridad, pero aparecen otros que necesitan nuevas defensas. Ian Goodfellow, un experto en IA que forma parte del equipo de Google Brain, explica: «En estos momentos se puede hacer casi todo lo malo que se nos pueda ocurrir hacerle a un modelo de aprendizaje automático […], y defenderlo es verdaderamente muy difícil».[151]

Aparte del problema inmediato de cómo defenderse de los ataques, la existencia de ejemplos antagónicos da más resonancia a la pregunta que he hecho antes: ¿qué están aprendiendo estas redes? En concreto, ¿qué están aprendiendo para que sea tan fácil engañarlas? O quizá más importante, ¿nos estamos engañando a nosotros mismos cuando pensamos que estas

redes han aprendido verdaderamente los conceptos que intentamos enseñarles?

A mi juicio, el problema fundamental es de comprensión. Veamos la figura 18, en la que AlexNet confunde un autobús escolar con un avestruz. ¿Por qué es tan improbable que le pase a un ser humano? Aunque AlexNet funciona muy bien en ImageNet, los humanos entendemos muchas cosas sobre los objetos que vemos que no saben ni AlexNet ni otros sistemas de IA actuales. Sabemos cómo son los objetos en tres dimensiones y podemos imaginárnoslos a partir de una foto bidimensional. Sabemos cuál es la función de un objeto determinado, qué papel desempeñan las partes del objeto en su función general y en qué contextos suele aparecer. Cuando vemos un objeto nos acordamos de haber visto otros iguales en distintas circunstancias y desde otros puntos de vista, además de haberlos percibido en otras modalidades sensoriales (recordamos el tacto de un objeto determinado, cómo huele, quizá cómo suena cuando se deja caer, etc.). Todos estos conocimientos previos contribuyen a la capacidad humana de identificar con claridad un objeto concreto. Incluso los mejores sistemas de visión artificial carecen de este tipo de conocimiento y de la solidez que eso les otorgaría.

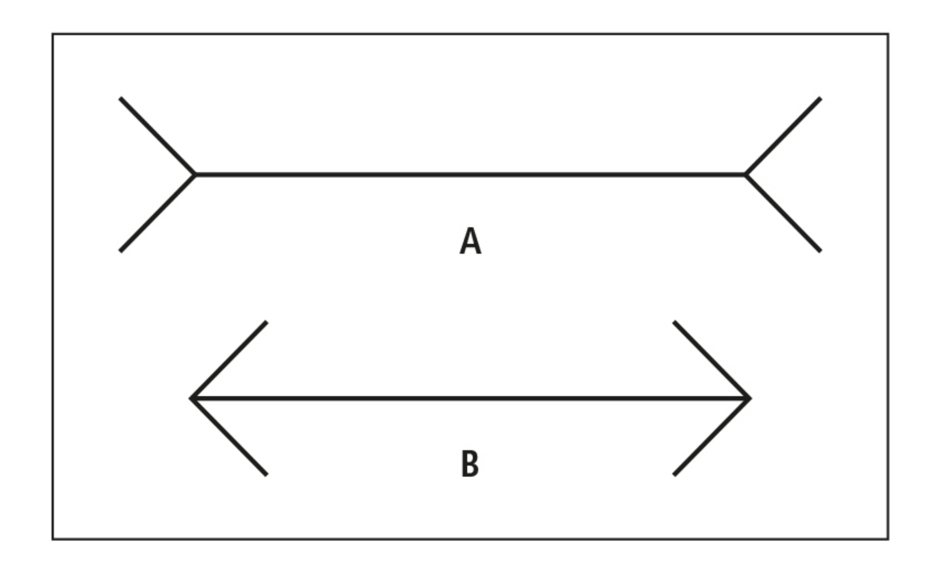


Figura 21. Una ilusión visual para los humanos: los segmentos de línea horizontal en A y B tienen la misma longitud, pero la mayoría de la gente percibe que el segmento en A es más largo que el de B.

He oído decir a algunos investigadores de IA que los humanos también somos vulnerables a nuestros propios tipos de «ejemplos antagónicos»: las ilusiones ópticas. Igual que AlexNet clasifica un autobús escolar como un avestruz, los humanos somos susceptibles de cometer errores de percepción (por ejemplo, nos parece que la línea superior de la figura 21 es más larga que la inferior, aunque en realidad ambas tienen la misma longitud). Pero los errores que cometemos los humanos son muy distintos de los que cometen las redes neuronales convolucionales: nuestra capacidad de reconocer objetos en escenas cotidianas ha evolucionado hasta ser muy sólida porque dependemos de ella para sobrevivir. A diferencia de las ConvNet actuales, la percepción humana (y animal) está muy regulada por la cognición, la comprensión basada en el contexto de la que he hablado antes. Además, las ConvNet que se utilizan hoy en día en las aplicaciones de visión por ordenador suelen ser totalmente de «prealimentación», mientras que el sistema visual humano tiene muchas más conexiones de «retroalimentación» (es decir, en dirección inversa) que de «prealimentación». Aunque los neurocientíficos aún no comprenden la función de toda esta retroalimentación, se podría aventurar que al menos algunas de esas conexiones de retroalimentación consiguen prevenir la vulnerabilidad a ejemplos antagónicos como los casos a los que son susceptibles las ConvNet. Si es así, ¿por qué no dar a las ConvNet el mismo tipo de retroalimentación? Es un área en la que se está investigando, pero es muy difícil y no ha tenido tanto éxito como las redes de prealimentación.

Jeff Clune, investigador de IA de la Universidad de Wyoming, hizo una analogía muy estimulante cuando señaló que hay un gran interés en saber si el aprendizaje profundo es «verdadera inteligencia» o un «Hans el Listo». [152] Hans el Listo fue un caballo alemán de principios del siglo XX que,

según su dueño, podía hacer cálculos aritméticos y entendía alemán. El caballo respondía a preguntas como «¿Cuánto es quince dividido por tres?» golpeando con la pezuña la cifra correcta. Después de que Hans el Listo se convirtiera en una celebridad internacional, una minuciosa investigación reveló que el caballo no entendía las preguntas ni los conceptos matemáticos que se le planteaban, sino que daba los golpes en función de unas señales sutiles que daba inconscientemente quien le preguntaba. Hans el Listo se ha convertido en una forma de llamar a cualquier individuo (o programa) que da la impresión de comprender pero que, en realidad, reacciona ante las señales involuntarias del entrenador. ¿El aprendizaje profundo tiene «verdadera comprensión» o es más bien un Hans el Listo informático que responde a señales superficiales encerradas en los datos? Esta duda es hoy objeto de acalorados debates en el mundo de la IA, con el agravante de que los investigadores de la IA no están necesariamente de acuerdo sobre la definición de «verdadera comprensión».

Por un lado, las redes neuronales profundas, entrenadas mediante aprendizaje supervisado, funcionan extraordinariamente bien (aunque todavía lejos de la perfección) en muchos problemas de visión por ordenador y en otros campos como el reconocimiento del habla y la traducción de idiomas. Estas redes, gracias a sus impresionantes capacidades, están saliendo rápidamente del mundo de la investigación para emplearse en aplicaciones del mundo real como la búsqueda en internet, los coches autónomos, el reconocimiento facial, los asistentes virtuales y los sistemas de recomendación, y cada vez resulta más difícil imaginar la vida sin estas herramientas de IA. Por otro lado, es engañoso decir que las redes profundas «aprenden solas» o que su entrenamiento es «similar al aprendizaje humano». Además de reconocer el éxito de estas redes, hay que matizar que pueden fallar de forma inesperada debido al sobreajuste a sus datos de entrenamiento, los efectos de cola larga y la vulnerabilidad a la piratería informática. Además, los motivos de las redes neuronales

profundas a la hora de tomar decisiones son muchas veces difíciles de entender, por lo que es difícil predecir y solucionar los fallos. Los investigadores trabajan sin cesar para que las redes neuronales profundas sean más fiables y transparentes, pero sigue habiendo una pregunta sin respuesta: si estos sistemas carecen de una comprensión similar a la humana, ¿es inevitable que sean frágiles, poco fiables y vulnerables a los ataques? ¿Y cómo debe influir eso en nuestras decisiones sobre la utilización de sistemas de IA en el mundo real? El próximo capítulo examina algunas de las formidables dificultades que entraña intentar encontrar el equilibrio entre los beneficios de la IA y los riesgos de su falta de fiabilidad y su uso indebido.

---

[125] Los lectores que siguieron las elecciones presidenciales estadounidenses de 2016 reconocerán el juego de palabras del eslogan de los partidarios de Bernie Sanders, «Feel the Bern».

[126] E. Brynjolfsson y A. McAfee, «The Business of Artificial Intelligence», *Harvard Business Review*, julio de 2017.

[127] O. Tanz, «Can Artificial Intelligence Identify Pictures Better than Humans?», *Entrepreneur*, 1 de abril de 2017, www.entrepreneur.com/article/283990.

[128] D. Vena, «3 Top AI Stocks to Buy Now», *Motley Fool*, 27 de marzo de 2017, www.fool.com/investing/2017/03/27/3-top-ai-stocks-to-buy-now.aspx.

[129] Citado en C. Metz, «A New Way for Machines to See, Taking Shape in Toronto», *The New York Times*, 28 de noviembre de 2017, www.nytimes.com/2017/11/28/technology/artificial-intelligence-research-toronto.html.

[130] Citado en J. Tanz, «Soon We Won't Program Computers. We'll Train Them Like Dogs», *Wired*, 17 de mayo de 2016.

[131] De la conferencia de Harry Shum en la Microsoft Faculty Summit, Redmond, Washington, junio de 2017.

[132] Este tema se analiza en profundidad en J. Lanier, *Who Owns the Future?*, Nueva York: Simon & Schuster, 2013.

[133] «Política de privacidad del cliente» de Tesla, consultada el 7 de diciembre de 2018, www.tesla.com/about/legal.

[134] T. Bradshaw, «Self-Driving Cars Prove to Be Labour-Intensive for Humans», *Financial Times*, 8 de julio de 2017.

[135] «Ground Truth Datasets for Autonomous Vehicles», Mighty AI, consultado el 7 de diciembre de 2018, mty.ai/adas/.

[136] «Deep Learning in Practice: Speech Recognition and Beyond», EmTech Digital video, 23 de mayo de 2016, events.technologyreview.com/emtech/digital/16/video/watch/andrew-ng-deep-learning.

[137] Y. Bengio, «Machines That Dream», en *The Future of Machine Intelligence: Perspectives from Leading Practitioners*, ed. de D. Beyer, Sebastopol, Calif.: O'Reilly Media, p. 14.

[138] W. Landecker *et al.*, «Interpreting Individual Classifications of Hierarchical Networks», en *Proceedings of the 2013 IEEE Symposium on Computational Intelligence and Data Mining* (2013), pp. 32-38.

[139] M. R. Loghmani *et al.*, «Recognizing Objects in-the-Wild: Where Do We Stand?», en *IEEE International Conference on Robotics and Automation* (2018), pp. 2170-2177.

[140] H. Hosseini *et al.*, «On the Limitation of Convolutional Neural Networks in Recognizing Negative Images», en *Proceedings of the 16th IEEE International Conference on Machine Learning and Applications* (2017), pp. 352-358; R. Geirhos *et al.*, «Generalisation in Humans and Deep Neural Networks», *Advances in Neural Information Processing Systems* 31 (2018), pp. 7549-7561; M. Alcorn *et al.*, «Strike (with) a Pose: Neural Networks Are Easily Fooled by Strange Poses of Familiar Objects», arXiv:1811.11553 (2018).

[141] M. Orcutt, «Are Face Recognition Systems Accurate? Depends on Your Race», *Technology Review*, 6 de julio de 2016, www.technologyreview.com/s/601786/are-facerecognition-systems-accurate-depends-on-your-race.

[142] J. Zhao *et al.*, «Men Also Like Shopping: Reducing Gender Bias Amplification Using Corpus-Level Constraints», en *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (2017).

[143] W. Knight, «The Dark Secret at the Heart of AI», *Technology Review*, 11 de abril de 2017, www.technologyreview.com/s/604087/the-dark-secret-at-the-heart-of-ai/.

[144] C. Szegedy *et al.*, «Intriguing Properties of Neural Networks», en *Proceedings of the International Conference on Learning Representations* (2014).

[145] A. Nguyen, J. Yosinski y J. Clune, «Deep Neural Networks Are Easily Fooled: High Confidence Predictions for Unrecognizable Images», en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 427-436.

[146] Véase, por ejemplo, M. Mitchell, *An Introduction to Genetic Algorithms*, Cambridge, Mass.: MIT Press, 1996.

[147] Nguyen, Yosinski y Clune, «Deep Neural Networks Are Easily Fooled».

[148] M. Sharif *et al.*, «Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition», en *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (2016), pp. 1528-1540.

[149] K. Eykholt *et al.*, «Robust Physical-World Attacks on Deep Learning Visual Classification», en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 1625-1634.

[150] S. G. Finlayson *et al.*, «Adversarial Attacks on Medical Machine Learning», *Science* 363, n.º 6433 (2019), pp. 1287-1289.

[151] Citado en W. Knight, «How Long Before AI Systems Are Hacked in Creative New Ways?», *Technology Review*, 15 de diciembre de 2016, www.technologyreview.com/s/603116/how-long-before-ai-systems-are-hacked-in-creative-new-ways.

[152] J. Clune, «How Much Do Deep Neural Networks Understand About the Images They Recognize?», diapositivas de conferencias (2016), consultadas el 7 de diciembre de 2018, c4dm.eecs.qmul.ac.uk/horse2016/HORSE2016Clune.pdf.

07

Sobre una IA ética y de confianza

Imagine que va en un coche autónomo, a altas horas de la noche, después de la fiesta de Navidad de la oficina. Está oscuro y nevando. «Coche, llévame a casa», dice, cansado y un poco bebido. Se recuesta en el asiento y da las gracias por poder cerrar los ojos mientras el coche arranca y se incorpora al tráfico.

Estupendo, pero ¿hasta qué punto se puede sentir seguro? La eficacia de los coches autónomos depende enormemente del aprendizaje automático (en especial del aprendizaje profundo), sobre todo en relación con los componentes de visión por ordenador y toma de decisiones. ¿Cómo podemos saber si estos coches han aprendido todo lo que necesitan saber?

Esta es la pregunta crucial para la industria de los coches autónomos. Los expertos tienen opiniones contradictorias sobre cuánto tiempo tardarán los vehículos autónomos en desempeñar un papel importante en nuestra vida cotidiana, con predicciones que oscilan (en el momento de escribir esto) entre unos cuantos años y muchas décadas. Los coches autónomos pueden mejorar de forma increíble nuestra vida. Los vehículos automatizados pueden reducir sustancialmente los millones de muertos y heridos anuales por accidentes de tráfico, muchos de ellos causados por conductores ebrios o distraídos. Además, los vehículos autónomos permitirían que los

pasajeros fueran productivos en lugar de estar inactivos durante el trayecto al trabajo. Asimismo, estos vehículos tienen el potencial de ser más eficientes energéticamente que los coches conducidos por humanos, y serán una bendición para las personas ciegas o discapacitadas que no pueden conducir. Pero todo esto solo se materializará si los humanos estamos dispuestos a poner nuestra vida en manos de estos vehículos.

El aprendizaje automático se está utilizando para tomar decisiones que afectan a la vida de los seres humanos en muchos ámbitos. ¿Qué garantías tenemos de que las máquinas que crean las noticias, diagnostican las enfermedades, evalúan nuestras solicitudes de préstamo o —Dios no lo quiera— recomiendan que nos condenen a prisión han aprendido lo suficiente como para tomar decisiones fiables?

Se trata de preguntas incómodas no solo para los investigadores de IA, sino también para la sociedad en su conjunto, que tendrá que sopesar tanto los múltiples usos positivos de la IA hoy y en el futuro como las preocupaciones acerca de su fiabilidad y su uso indebido.

La IA beneficiosa

Cuando reflexionamos sobre el papel de la IA en nuestra sociedad, quizá es fácil pensar solo en los aspectos negativos. Sin embargo, es esencial recordar que los sistemas de IA ya benefician mucho a la sociedad y tienen posibilidades de hacerlo aún más. La tecnología de IA actual es fundamental para servicios que utilizamos todo el tiempo, a veces sin saber que esta interviene, como la transcripción de voz, la navegación y la planificación de trayectos por GPS, los filtros de *spam* de correo electrónico, la traducción de idiomas, las alertas de fraude con tarjetas de crédito, las recomendaciones de libros y música, la protección contra los virus informáticos y la optimización del consumo de energía en los edificios.

Si usted es fotógrafo, cineasta, artista o músico es probable que utilice sistemas de IA que le facilitan sus proyectos creativos, como los programas que ayudan a los fotógrafos a editar sus fotos o a los compositores en la notación o los arreglos musicales. Un estudiante puede sacar partido de los «sistemas de tutoría inteligente» que se adaptan a su estilo de aprendizaje. Un científico seguramente habrá utilizado alguna de las muchas herramientas de IA disponibles que facilitan el análisis de datos. Una persona ciega o con algún tipo de discapacidad visual puede utilizar aplicaciones de visión por ordenador para teléfonos móviles que leen texto manuscrito o impreso (por ejemplo, los textos de los carteles, los menús de restaurante o el dinero). Alguien con problemas de audición dispone ahora de subtítulos muy exactos en los vídeos de YouTube y, en algunos casos, de transcripciones de voz en tiempo real durante una conferencia. Estos no son más que unos cuantos ejemplos de las mejoras que las actuales herramientas de IA están introduciendo en la vida de las personas. Hay muchas otras tecnologías de IA que todavía están en fase de investigación, pero a punto de generalizarse.

En un futuro próximo, las aplicaciones de la IA probablemente se generalizarán en la atención sanitaria. Los sistemas de IA ayudarán a los médicos a diagnosticar enfermedades y sugerir tratamientos, a descubrir nuevos fármacos y a controlar la salud y la seguridad de los ancianos en su hogar. La creación de modelos científicos y el análisis de datos se basarán cada vez más en herramientas de IA; por ejemplo, para mejorar los modelos de cambio climático, crecimiento y cambio demográfico, ciencias medioambientales y de la alimentación y otros problemas importantes que afrontará la sociedad durante el próximo siglo. Para Demis Hassabis, cofundador del grupo DeepMind de Google, este es el posible beneficio más importante de la IA:

> Puede que tengamos que hacernos a la idea de que, incluso aunque los seres humanos más inteligentes del planeta estén trabajando en estos problemas, estos [problemas] son quizá tan complejos que es difícil que a las personas y los expertos científicos, de forma individual, les

> dé tiempo en la vida para innovar y avanzar… En mi opinión, vamos a necesitar ayuda, y creo que la IA es la solución.[153]

Todos hemos oído decir que en el futuro la inteligencia artificial se hará cargo de los trabajos que los humanos detestan: trabajos mal pagados, aburridos, agotadores, degradantes, explotadores o directamente peligrosos. Si es así, podría ser una verdadera bendición para el bienestar humano. (Más adelante hablaré de la otra cara de la moneda: que la inteligencia artificial elimine demasiados puestos de trabajo humanos). Los robots ya se utilizan mucho en las fábricas para tareas no cualificadas y repetitivas, aunque hay muchos trabajos de este tipo que aún superan las aptitudes de los robots actuales. Pero, a medida que avance la IA, la automatización podrá hacerse cada vez más cargo de estos trabajos. Entre los ejemplos de futuras aplicaciones de la IA en el lugar de trabajo están los camiones y taxis autónomos y el uso de robots para la recolección de frutas, la extinción de incendios, la retirada de minas terrestres y la limpieza del medio ambiente. Además, los robots tendrán con toda probabilidad un papel aún mayor que el actual en la exploración planetaria y espacial.

¿Será verdaderamente beneficioso para la sociedad que los sistemas de IA se hagan cargo de esos trabajos? La historia de la tecnología nos puede ayudar a poner las cosas en perspectiva. Veamos algunos ejemplos de puestos de trabajo que ocupaban los humanos pero que la tecnología automatizó hace tiempo, al menos en los países desarrollados: lavadora de ropa; conductor de *rickshaw* (bicitaxi); ascensorista; *punkawallah* (en la India, un criado que solo tenía la función de manejar un ventilador manual para enfriar la habitación, antes de que existieran los ventiladores eléctricos); calculadora (una persona, normalmente mujer, que hacía tediosos cálculos a mano, sobre todo durante la Segunda Guerra Mundial). Casi todo el mundo estará de acuerdo en que en esos casos sustituir a los humanos por máquinas mejoró la vida en general. Se podría decir que la IA actual no hace más que prolongar esa misma trayectoria de progreso:

mejorar la vida de los humanos automatizando cada vez más los trabajos necesarios que nadie quiere hacer.

El gran dilema sobre las contrapartidas de la AI

El investigador de IA Andrew Ng ha proclamado con optimismo: «La IA es la nueva electricidad». Ng explica, además: «Así como la electricidad transformó casi todo hace cien años, hoy me cuesta imaginar un sector que la IA no vaya a transformar en los próximos años».[154] Es una analogía atractiva: la idea de que pronto la IA será tan necesaria —y tan invisible— en nuestros dispositivos electrónicos como la propia electricidad. Sin embargo, una diferencia importante es que las bases científicas de la electricidad se conocían bien ya antes de que empezara su comercialización. Sabemos predecir cómo se va a comportar la electricidad. Eso no pasa con muchos de los sistemas de IA actuales.

Esto nos lleva a lo que podríamos llamar el gran dilema sobre las contrapartidas de la IA. ¿Debemos aceptar las capacidades de los sistemas de IA, que pueden mejorar nuestras vidas e incluso ayudar a salvarlas, y permitir que se empleen cada vez más? ¿O tenemos que ser más cautos, dados los errores impredecibles, la susceptibilidad a los sesgos, la vulnerabilidad a los ataques informáticos y la falta de transparencia de las decisiones que encontramos en la IA actual? ¿Hasta qué punto debe haber siempre humanos al tanto de todo en las distintas aplicaciones de la IA? ¿Qué debemos exigir a un sistema de IA para confiar en que es capaz de funcionar de forma autónoma? Estas cuestiones siguen siendo objeto de acalorados debates, al tiempo que la IA se extiende cada vez más y se nos asegura que las aplicaciones más prometedoras (por ejemplo, los coches autónomos) están a la vuelta de la esquina.

La falta de consenso sobre estas cuestiones quedó patente en un estudio reciente del Pew Research Center.[155] En 2018, los analistas de Pew hicieron una encuesta entre casi mil «pioneros tecnológicos, innovadores,

desarrolladores, dirigentes empresariales y políticos, investigadores y activistas», a los que pidieron que respondieran a estas preguntas:

> De aquí a 2030, ¿cree que lo más probable es que los avances de la IA y los sistemas tecnológicos relacionados mejoren y refuercen las capacidades humanas? Es decir, ¿la mayoría de las personas estarán la mayor parte del tiempo mejor que ahora? ¿O es más probable que los avances de la IA y los sistemas tecnológicos relacionados reduzcan la autonomía y la capacidad de acción de los humanos hasta el punto de que la mayoría de las personas no estarán mejor que hoy?

Hubo división de respuestas. El 63 por ciento predijo que los avances de la IA crearían una situación mejor para los humanos de aquí a 2030, mientras que el 37 por ciento se mostró en desacuerdo. Las respuestas oscilaban entre la opinión de que la IA «puede eliminar prácticamente toda la pobreza mundial, reducir enormemente las enfermedades y proporcionar mejor educación a casi todos los habitantes del planeta» hasta predicciones de un futuro apocalíptico: legiones de puestos de trabajo anulados por la automatización, merma de la privacidad y los derechos civiles por las funciones de vigilancia de la IA, armas autónomas amorales, decisiones incontroladas de programas informáticos opacos y poco fiables, intensificación de los prejuicios raciales y de género, manipulación de los medios de comunicación, aumento de la ciberdelincuencia y lo que uno de los encuestados denominó la «auténtica irrelevancia existencial» para los humanos.

La inteligencia artificial presenta un enrevesado abanico de dudas éticas; los debates relacionados con la ética de la IA y los macrodatos han llenado ya varios libros.[156] Para ilustrar la complejidad de estas cuestiones, voy a profundizar en un ejemplo que está siendo objeto de mucha atención en los últimos tiempos: el reconocimiento facial automático.

La ética del reconocimiento facial

El reconocimiento facial es la tarea de etiquetar un rostro en una imagen o un vídeo (o una retransmisión por directo) con un nombre. Facebook, por ejemplo, aplica un algoritmo de reconocimiento facial a todas las fotos que se suben a su web para identificar las caras de la foto y emparejarlas con usuarios conocidos (al menos los usuarios que no han desactivado esta función).[157] Cuando una persona está en Facebook y alguien publica una foto que incluye su cara, el sistema puede preguntarle si quiere «etiquetarse» en la foto. La precisión del algoritmo de reconocimiento facial de Facebook puede ser al mismo tiempo admirable y escalofriante. No es extraño que esa precisión se deba al uso de redes neuronales convolucionales profundas. Muchas veces, el programa puede identificar un rostro en la foto no solo cuando está en primer plano, sino incluso cuando esa persona está en medio de una multitud.

La tecnología de reconocimiento facial tiene muchas posibles ventajas, como ayudar a la gente a buscar entre sus colecciones de fotos, hacer posible a las personas con problemas de visión identificar a las personas con las que se encuentran, localizar a niños desaparecidos o delincuentes fugitivos escaneando fotos y vídeos para identificarlos, y detectar robos de identidad. Sin embargo, también es fácil imaginar aplicaciones que resulten ofensivas o amenazadoras para muchas personas. Amazon, por ejemplo, ha puesto a la venta su sistema de reconocimiento facial (con el peculiar y distópico nombre de Rekognition) dirigido a departamentos de policía, que, por ejemplo, pueden comparar grabaciones de cámaras de seguridad con una base de datos de delincuentes conocidos o sospechosos probables.

Un problema evidente es la privacidad. Incluso aunque una persona no esté en Facebook (ni en cualquier otra plataforma de redes sociales con reconocimiento facial), se pueden etiquetar las fotos en las que aparece para reconocerlas después de forma automática y sin su permiso. Pensemos en FaceFirst, una empresa que ofrece servicios de pago de reconocimiento facial. La revista *New Scientist* informa de que «Face First […] está

poniendo en marcha un sistema para comercios minoristas que, según asegura, “impulsará las ventas al reconocer a los clientes más valorados cada vez que compren” y enviará “alertas cuando personas conocidas por ser conflictivas entren en cualquiera de sus establecimientos”».[158] Muchas otras empresas ofrecen servicios similares.

La pérdida de privacidad no es el único peligro. Una preocupación aún mayor es la fiabilidad: los sistemas de reconocimiento facial pueden cometer errores. Si identifican equivocadamente la cara de una persona como si fuera de otra, pueden prohibirle el acceso a una tienda o a un vuelo, o acusarle sin razón de un delito. Además, se ha demostrado que los sistemas actuales de reconocimiento facial tienen un porcentaje de error considerablemente mayor con las personas de color que en las blancas. La Unión Estadounidense de Libertades Civiles (ACLU, por sus siglas en inglés), que, en su defensa de los derechos civiles, se opone enérgicamente a que las fuerzas del orden utilicen la tecnología de reconocimiento facial, probó el sistema Rekognition de Amazon (con su configuración predeterminada) con los 535 miembros del Congreso de Estados Unidos, comparando una foto de cada uno de ellos con una base de datos de personas que habían sido detenidas por cargos penales. Descubrieron que el sistema había emparejado equivocadamente a 28 de los 535 congresistas con personas de la base de datos de delincuentes. El 21 por ciento de los errores eran en fotos de representantes afroamericanos (que solo constituyen el 9 por ciento del Congreso).[159]

En vista de los resultados de las pruebas de la ACLU y de otros estudios que demuestran la falta de fiabilidad y los sesgos del reconocimiento facial, varias empresas de alta tecnología han anunciado que se oponen al uso del reconocimiento facial por parte de las fuerzas del orden y en tareas de vigilancia. Por ejemplo, Brian Brackeen, director ejecutivo de la empresa de reconocimiento facial Kairos, escribió lo siguiente en un artículo muy difundido:

> Las tecnologías de reconocimiento facial, utilizadas en la identificación de sospechosos, perjudican a las personas de color. Negarlo sería mentir […]. Mi empresa y yo hemos llegado a la conclusión de que el uso del reconocimiento facial comercial para fines policiales o en vigilancia gubernamental de cualquier tipo es malo y abre la puerta a graves faltas de ética por parte de personas moralmente corruptas […]. Nos merecemos un mundo en el que no permitamos a los Gobiernos clasificar, seguir la pista y controlar a los ciudadanos.[160]

En una entrada del blog de la web de su empresa, el presidente y director jurídico de Microsoft, Brad Smith, pidió al Congreso que regulara el reconocimiento facial:

> La tecnología de reconocimiento facial plantea dudas que afectan directamente a la protección de derechos humanos fundamentales como la privacidad y la libertad de expresión. Estas cuestiones aumentan la responsabilidad de las empresas tecnológicas que crean estos productos. En nuestra opinión, también exigen una regulación gubernamental prudente y el desarrollo de normas que indiquen qué usos son aceptables. Con el reconocimiento facial, tanto el sector público como el privado deben dar un paso al frente y actuar.[161]

Google siguió su ejemplo y anunció que no ofrecería servicios de reconocimiento facial en general a través de su plataforma de IA en la nube hasta que la empresa pudiera «garantizar que su uso coincide con nuestros principios y valores, y evita malos usos y resultados perjudiciales».[162]

La reacción de estas empresas invita al optimismo, pero pone sobre la mesa otra cuestión fastidiosa: ¿hasta qué punto deben regularse la investigación y el desarrollo de la IA, y quién debe hacerlo?

La regulación de la IA

Dados los riesgos de las tecnologías de IA, muchos profesionales del sector, entre los que me incluyo, piensan que debe haber algún tipo de regulación. Pero la regulación no debe dejarse únicamente en manos de los investigadores y las empresas de IA. Los problemas relacionados con la IA —la fiabilidad, la necesidad de explicaciones, el sesgo, la vulnerabilidad a los ataques y los aspectos morales— son cuestiones no solo técnicas, sino también sociales y políticas. Por tanto, es esencial que en el debate

intervengan personas con diferentes perspectivas y procedencias. Dejar la regulación en manos de los profesionales de la IA sería tan imprudente como dejarla exclusivamente en manos de los organismos gubernamentales.

Como ejemplo de lo compleja que es la elaboración de este tipo de reglamentos, en 2018 el Parlamento Europeo promulgó un reglamento sobre IA que algunos han denominado el «derecho a la explicación».[163] Este reglamento exige que en caso de «decisiones automatizadas», haya «información de peso sobre la lógica involucrada» en cualquier decisión que afecte a un ciudadano de la UE. Esta información debe comunicarse «de forma concisa, transparente, inteligible y fácilmente accesible, con un lenguaje claro y sencillo».[164] La frase abre la puerta a la interpretación. ¿Qué es «información de peso» o «la lógica involucrada»? ¿Prohíbe esta normativa el uso de métodos de aprendizaje profundo difíciles de explicar para tomar decisiones que afectan a las personas (como los préstamos y el reconocimiento facial)? No hay duda de que estas incertidumbres garantizarán un empleo remunerado para los políticos y los abogados durante mucho tiempo.

Creo que la regulación de la IA debe seguir el modelo de la de otras tecnologías, en particular las de las ciencias biológicas y médicas, como la ingeniería genética. En esos campos, las normas —por ejemplo, las garantías de calidad y el análisis de riesgos y beneficios de las tecnologías — se elaboran mediante la cooperación entre organismos gubernamentales, empresas, organizaciones sin ánimo de lucro y universidades. Además, ya existen campos consolidados de la bioética y la ética médica que influyen de forma considerable en las decisiones sobre el desarrollo y la aplicación de las tecnologías. La investigación de la IA y sus aplicaciones necesitan sobre todo una infraestructura reguladora y ética muy meditada.

Esta infraestructura está empezando a formarse. En Estados Unidos, los Gobiernos estatales están empezando a estudiar la creación de normativas, entre ellas sobre el reconocimiento facial y los vehículos autónomos. Ahora

bien, en general, se ha dejado que las universidades y las empresas que crean sistemas de IA se autorregulen.

Para llenar este vacío han surgido varios grupos de reflexión sin ánimo de lucro, con frecuencia financiados por ricos empresarios tecnológicos preocupados por la IA. Estas organizaciones —con nombres como Future of Humanity Institute, Future of Life Institute y Centre for the Study of Existential Risk— organizan seminarios, patrocinan investigaciones y elaboran materiales educativos y sugerencias políticas sobre los usos seguros y éticos de la IA. Una organización marco llamada Partnership on AI ha intentado reunir a estos grupos para «ser una plataforma abierta de debate y diálogo sobre la IA y su influencia en las personas y la sociedad». [165]

Uno de los escollos es que no hay consenso en este campo sobre cuáles deben ser las prioridades a la hora de desarrollar la regulación y las normas éticas. ¿Debemos centrarnos ante todo en unos algoritmos que puedan explicar su razonamiento? ¿En la privacidad de los datos? ¿En la solidez de los sistemas de IA frente a los ataques maliciosos? ¿En los sesgos de los sistemas de IA? ¿En el posible «riesgo existencial» que representa una IA superinteligente? En mi opinión, se ha prestado demasiada atención a los riesgos de la IA superinteligente y muy poca tanto a la falta de fiabilidad y transparencia del aprendizaje profundo como a su vulnerabilidad a los ataques. Hablaré más sobre la idea de superinteligencia en el último capítulo.

Máquinas morales

Hasta ahora, mi análisis se ha centrado en las dudas éticas sobre cómo utilizan los humanos la IA. Pero hay otra cuestión importante: ¿podrían las máquinas tener su propio sentido moral, tanto como para que les permitamos tomar decisiones éticas por su cuenta, sin que los humanos tengan que supervisarlas? Si vamos a dar autonomía de decisión a los

sistemas de reconocimiento facial, a los coches autónomos, a los robots que cuidan ancianos o incluso a los robots militares, ¿no deberíamos darles la misma capacidad de abordar cuestiones éticas y morales que tenemos los humanos?

El «sentido moral de las máquinas» es objeto de reflexión desde hace tanto tiempo como la IA.[166] Probablemente, el debate más conocido sobre la ética de las máquinas es el de los relatos de ciencia ficción de Isaac Asimov, que proponía las tres «leyes fundamentales de la robótica»:

1. Un robot no puede dañar a un ser humano ni, por inacción, permitir que un ser humano sufra daños.
2. Un robot debe obedecer las órdenes que le den los seres humanos, excepto cuando tales órdenes entren en conflicto con la Primera Ley.
3. Un robot debe proteger su propia existencia siempre que dicha protección no entre en conflicto con la Primera o la Segunda Ley.[167]

Estas leyes son famosas, pero, en realidad, el propósito de Asimov era demostrar que una serie de reglas de este tipo no tenía más remedio que fracasar. «Círculo vicioso», el relato de 1942 en el que Asimov presentó por primera vez estas leyes, plantea una situación en la que un robot, por obedecer la segunda ley, se acerca a una sustancia peligrosa, momento en el que entra en vigor la tercera ley, así que el robot se aleja, momento en el que vuelve a entrar en vigor la segunda ley, de modo que el robot queda atrapado en un bucle sin fin, lo que tiene consecuencias casi desastrosas para los amos humanos del robot. Los relatos de Asimov solían fijarse en las consecuencias imprevistas de programar reglas éticas en los robots. Y era profético: como hemos visto, el problema de las reglas incompletas y las consecuencias imprevistas ha sido un estorbo para todas las formas de enfocar la inteligencia de la IA basada en reglas; lo mismo pasa con el razonamiento moral.

El escritor de ciencia ficción Arthur C. Clarke utilizó un recurso argumental similar en su libro de 1968 *2001, una odisea espacial*.[168] El ordenador con inteligencia artificial HAL está programado para decir

siempre la verdad a los humanos, pero, al mismo tiempo, también para ocultar a los astronautas humanos el propósito real de su misión espacial. HAL, a diferencia del robot despistado de Asimov, sufre las consecuencias psicológicas de esta disonancia cognitiva: «Era […] consciente del conflicto que estaba destruyendo poco a poco su integridad: el conflicto entre la verdad y la ocultación de la verdad».[169] El resultado es una «neurosis» informática que convierte a HAL en asesino. Reflexionando sobre la moralidad de las máquinas en la vida real, el matemático Norbert Wiener señaló ya en 1960 que «más nos vale estar muy seguros de que el propósito que se introduce en la máquina es el propósito que verdaderamente deseamos».[170]

El comentario de Wiener capta lo que se denomina el problema de la alineación de valores en la IA: la necesidad de que los programadores de IA puedan garantizar que los valores de sus sistemas coinciden con los de los humanos. ¿Pero cuáles son los valores humanos? ¿Tiene sentido siquiera suponer que existen unos valores universales que comparte toda la sociedad?

Bienvenidos a Filosofía Moral I. Empezaremos con el experimento teórico favorito de todo estudiante de Filosofía Moral, el dilema del tranvía. Una persona conduce un tranvía que va a toda velocidad y de pronto ve delante a cinco trabajadores en plena vía. Pisa el freno, pero se da cuenta de que no funciona. Por suerte, hay un ramal que sale hacia la derecha. Si lleva el tranvía hacia el ramal, no atropellará a los cinco trabajadores. Pero en medio de esa otra vía hay otro trabajador, solo uno. Si no se desvía, el tranvía atropellará a los cinco trabajadores y los matará a todos. Si se va hacia la derecha, matará al trabajador que está solo. ¿Qué nos dicta la moral?

El dilema del tranvía es un elemento omnipresente en los cursos universitarios de ética desde hace un siglo. La mayoría de la gente responde que, desde el punto de vista moral, es preferible que el conductor se desvíe

a la otra vía, mate al trabajador que está solo y salve al grupo de los cinco. Pero los filósofos han descubierto que un enunciado distinto del mismo dilema puede llevar a la gente a responder todo lo contrario.[171] Parece que los razonamientos humanos sobre dilemas morales son muy sensibles a cómo se los presentan.

El dilema del tranvía ha reaparecido en los últimos tiempos dentro de las informaciones mediáticas sobre los vehículos autónomos,[172] y la cuestión de cómo debe programarse un vehículo autónomo para afrontar este tipo de problemas se ha convertido en un tema de conversación fundamental de los debates sobre la ética de la IA. Muchos teóricos de la ética de la IA destacan que el dilema del tranvía propiamente dicho, en el que el conductor no tiene más que dos opciones horribles, es una situación muy forzada que ningún conductor del mundo real se encontrará jamás. Pero el dilema se ha convertido en una especie de símbolo que nos obliga a preguntarnos cómo debemos programar los coches autónomos para que tomen decisiones morales por su cuenta.

En 2016, tres investigadores publicaron los resultados de unas encuestas llevadas a cabo entre varios centenares de personas a las que se les plantearon situaciones similares a las del dilema del tranvía en las que estaban implicados coches autónomos, y se les preguntó su opinión sobre diferentes acciones desde el punto de vista moral. En una de las encuestas, el 76 por ciento de los participantes respondió que sería moralmente preferible para un coche autónomo sacrificar a un pasajero que matar a diez peatones. Pero cuando se les preguntó si comprarían un coche autónomo programado para sacrificar a sus pasajeros si eso suponía salvar a un número mucho mayor de peatones, la inmensa mayoría de los encuestados respondieron que ellos no se lo comprarían.[173] Según los autores, «descubrimos que los participantes en seis estudios de Amazon Mechanical Turk aprobaban los AV [vehículos autónomos] utilitaristas (es decir, los AV que sacrificasen a sus pasajeros por un bien mayor) y les gustaría que otros

los compraran, pero ellos preferirían viajar en AV que protegieran a sus pasajeros a toda costa». En su comentario sobre este estudio, el psicólogo Joshua Greene señaló: «Antes de poder introducir nuestros valores en las máquinas, debemos aprender a hacer que nuestros valores sean claros y coherentes».[174] Algo que parece más difícil de lo que tal vez pensábamos.

Algunos investigadores de la ética de la IA han sugerido que dejemos de intentar programar directamente reglas morales para las máquinas y que, en su lugar, hagamos que las máquinas aprendan valores morales por sí mismas observando el comportamiento humano.[175] Pero ese enfoque autodidacta hereda todos los problemas del aprendizaje automático que he descrito en el capítulo anterior.

En mi opinión, los avances en la capacidad de dotar a los ordenadores de inteligencia moral no pueden separarse de los avances en otros tipos de inteligencia: el objetivo verdaderamente difícil es crear máquinas capaces de comprender verdaderamente las situaciones a las que se enfrentan. Como demuestran los relatos de Isaac Asimov, no podemos fiarnos de que un robot vaya a obedecer la orden de evitar hacer daño a un humano si no puede entender el concepto de daño en diferentes situaciones. Para razonar sobre moral hace falta identificar las relaciones causa-efecto, imaginar diferentes futuros posibles, tener cierta idea de las creencias y los propósitos de los demás y predecir los resultados probables de nuestras acciones en cualquier situación en la que nos encontremos. En otras palabras, un razonamiento moral digno de confianza necesita antes el sentido común general, que, como hemos visto, falta incluso en los mejores sistemas de IA actuales.

Hasta ahora hemos visto que las redes neuronales profundas, entrenadas con enormes conjuntos de datos, pueden rivalizar con las capacidades visuales de los humanos en determinadas tareas. También hemos visto algunos puntos débiles de esas redes, como su necesidad de cantidades ingentes de datos etiquetados por humanos y su propensión a fallar de

formas muy poco humanas. ¿Cómo podemos crear un sistema de IA que aprenda de verdad por sí solo, que sea más fiable porque, como los humanos, sea capaz de razonar sobre su situación actual y planificar el futuro? En la siguiente parte del libro voy a describir de qué forma los investigadores de IA están utilizando juegos como el ajedrez, el go e incluso los videojuegos de Atari como «microcosmos» para desarrollar máquinas con capacidades de aprendizaje y razonamiento más parecidas a las humanas, y voy a evaluar de qué manera esas máquinas jugadoras sobrehumanas podrían transferir sus aptitudes al mundo real.

---

[153] Citado en D. Palmer, «AI Could Help Solve Humanity's Biggest Issues by Taking Over from Scientists, Says DeepMind CEO», *Computing*, 26 de mayo de 2015, www.computing.co.uk/ctg/news/2410022/ai-could-help-solve-humanity-s-biggest-issuesby-taking-over-from-scientists-says-deepmind-ceo.

[154] S. Lynch, «Andrew Ng: Why AI Is the New Electricity», *Insights by Stanford Business*, 11 de marzo de 2017, www.gsb.stanford.edu/insights/andrew-ng-why-ainew-electricity.

[155] J. Anderson, L. Rainie y A. Luchsinger, «Artificial Intelligence and the Future Of Humans», Pew Research Center, 10 de diciembre de 2018, www.pewinternet.org/ 2018/12/10/artificial-intelligence-and-the-future-of-humans.

[156] Dos análisis recientes de las cuestiones éticas relacionadas con la IA y los macrodatos son C. O'Neil, *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*, Nueva York: Crown, 2016 [trad. cast.: *Armas de destrucción matemática. Cómo el* big data *aumenta la desigualdad y amenaza la democracia*, Madrid: Capitán Swing, 2018]; y H. Fry, *Hello World: Being Human in the Age of Algorithms*, Nueva York: W. W. Norton, 2018 [trad. cast.: *Hola mundo. Cómo seguir siendo humanos en la era de los algoritmos*, Barcelona: Blackie Books, 2019].

[157] C. Domonoske, «Facebook Expands Use of Facial Recognition to ID Users in Photos», National Public Radio, 19 de diciembre de 2017, www.npr.org/sections/thetwo-way/2017/12/19/571954455/facebook-expands-use-of-facial-recognition-toid-users-in-photos.

[158] H. Hodson, «Face Recognition Row over Right to Identify You in the Street», *New Scientist*, 19 de junio de 2015.

[159] J. Snow, «Amazon's Face Recognition Falsely Matched 28 Members of Congress with Mugshots», *Free Future* (blog), ACLU, 26 de julio de 2018, www.aclu.org/blog/privacy-technology/surveillance-technologies/amazons-face-recognition-falselymatched-28.

[160] B. Brackeen, «Facial Recognition Software Is Not Ready for Use by Law Enforcement», *Tech Crunch*, 25 de junio de 2018, techcrunch.com/2018/06/25/facialrecognition-software-is-not-

ready-for-use-by-law-enforcement.

[161] B. Smith, «Facial Recognition Technology: The Need for Public Regulation and Corporate Responsibility», *Microsoft on the Issues* (blog), Microsoft, 13 de julio de 2018, blogs.microsoft.com/on-the-issues/2018/07/13/facial-recognition-technology-theneed-for-public-regulation-and-corporate-responsibility.

[162] K. Walker, «AI for Social Good in Asia Pacific», *Around the Globe* (blog), Google, 13 de diciembre de 2018, www.blog.google/around-the-globe/google-asia/ai-social-good-asia-pacific.

[163] B. Goodman y S. Flaxman, «European Union Regulations on Algorithmic Decision-Making and a 'Right to Explanation' », *AI Magazine* 38, n.º 3 (otoño de 2017), pp. 50-57.

[164] «Article 12, EU GDPR: Transparent Information, Communication, and Modalities for the Exercise of the Rights of the Data Subject», Reglamento General de Protección de Datos de la UE, consultado el 7 de diciembre de 2018, www.privacy-regulation.eu/en/article-12-transparent-information-communication-and-modalitiesfor-the-exercise-of-the-rights-of-the-data-subject-GDPR.htm.

[165] Sitio web de Partnership on AI, consultado el 18 de diciembre de 2018, www.partnershiponai.org.

[166] Para un estudio más amplio de este tema, véase W. Wallach y C. Allen, *Moral Machines: Teaching Robots Right from Wrong*, Nueva York: Oxford University Press, 2008.

[167] I. Asimov, *I, Robot*, Bantam Dell, 2004, p. 37 (1.ª ed.: Grove, 1950) [trad. cast.: *Yo, robot*, Barcelona: Edhasa, 2019].

[168] A. C. Clarke, *2001: A Space Odyssey*, Londres: Hutchinson & Co, 1968 [trad. cast.: *2001, una odisea espacial*, Barcelona: Debolsillo, 2003].

[169] *Ibid.*, p. 192.

[170] N. Wiener, «Some Moral and Technical Consequences of Automation», *Science* 131, n.º 3410 (1960), pp. 1355-1358.

[171] J. J. Thomson, «The Trolley Problem», *Yale Law Journal* 94, n.º 6 (1985), pp. 1395-1415.

[172] Véase, por ejemplo, J. Achenbach, «Driverless Cars Are Colliding with the Creepy Trolley Problem», *The Washington Post*, 29 de diciembre de 2015.

[173] J.-F. Bonnefon, A. Shariff e I. Rahwan, «The Social Dilemma of Autonomous Vehicles», *Science* 352, n.º 6293 (2016), pp. 1573-1576.

[174] J. D. Greene, «Our Driverless Dilemma», *Science* 352, n.º 6293 (2016), pp. 1514-1515.

[175] Véase, por ejemplo, M. Anderson y S. L. Anderson, «Machine Ethics: Creating an Ethical Intelligent Agent», *AI Magazine* 28, n.º 4 (2007), p. 15.

PARTE III

APRENDAMOS A JUGAR

08

Recompensas para los robots

Cuando la periodista Amy Sutherland investigaba para un libro sobre adiestradores de animales exóticos, se enteró de que su método principal es absurdamente sencillo: «recompensar el comportamiento que me gusta e ignorar el que no». Y como escribió en la columna «Modern Love» de *The New York Times*: «Al final se me ocurrió que las mismas técnicas podían funcionar con esa especie cabezota pero adorable que es el marido estadounidense». Sutherland contó que, después de años de quejas inútiles, sarcasmo y resentimiento, utilizó ese sencillo método para entrenar subrepticiamente a su inconsciente marido con el fin de que recogiera los calcetines, encontrara las llaves del coche, llegara puntual a los restaurantes y se afeitara con más regularidad.[176]

Esta técnica clásica de adiestramiento, conocida en psicología como condicionamiento instrumental, se utiliza desde hace siglos con animales y seres humanos. El condicionamiento instrumental inspiró un importante método de aprendizaje automático llamado aprendizaje por refuerzo, que es distinto del método de aprendizaje supervisado que he descrito en capítulos anteriores: en su forma más pura, el aprendizaje por refuerzo no necesita ejemplos etiquetados de entrenamiento. En lugar de ello, un agente —el programa de aprendizaje— lleva a cabo acciones en un entorno

(normalmente una simulación por ordenador) y de vez en cuando recibe recompensas de ese entorno. Estas recompensas intermitentes son las únicas valoraciones que el agente utiliza para aprender. En el caso del marido de Amy Sutherland, las recompensas eran las sonrisas, los besos y las palabras de elogio. Un programa informático no reacciona ante un beso o un entusiasta «eres el mejor», pero se puede hacer que reaccione ante el equivalente informático de un «gracias»; por ejemplo, más números positivos añadidos a su memoria.

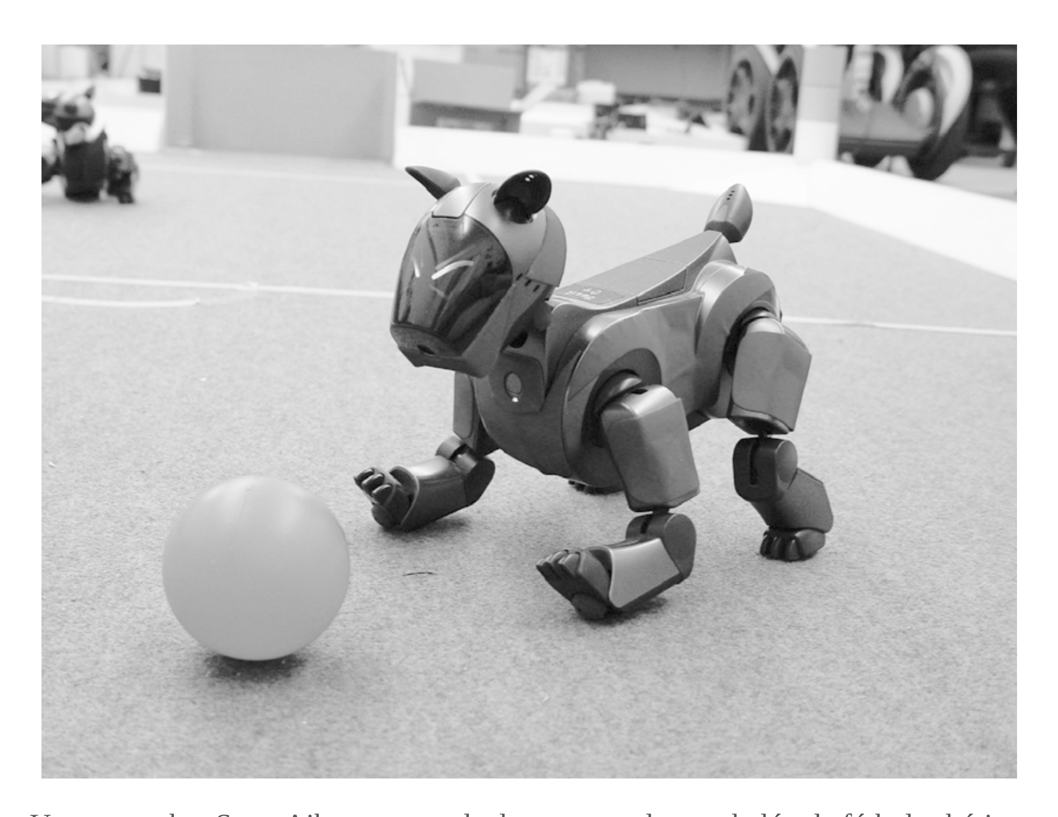
Aunque el aprendizaje por refuerzo es una de las herramientas utilizadas con la IA desde hace décadas, durante mucho tiempo quedó eclipsado por las redes neuronales y otros métodos de aprendizaje supervisado. La situación cambió en 2016, cuando el aprendizaje por refuerzo contribuyó de manera fundamental a un avance asombroso y trascendental de la IA: un programa que aprendió a vencer a los mejores campeones humanos del complejo juego del go. Para explicar ese programa y otros éxitos recientes del aprendizaje por refuerzo, antes tengo que mostrar un ejemplo sencillo para ilustrar cómo funciona el aprendizaje por refuerzo.

Cómo entrenar a su perro robot

Como ejemplo ilustrativo, veamos el divertido juego del fútbol robótico, en el que unos humanos (normalmente estudiantes universitarios) programan robots para que jueguen una versión simplificada del fútbol en un «campo» del tamaño de una habitación. A veces, los jugadores son simpáticos robots Aibo con forma de perro, como el de la figura 22. Un robot Aibo (fabricado por Sony) tiene una cámara para captar imágenes, un ordenador interno programable y un conjunto de sensores y motores que le permiten andar, dar patadas, dar cabezazos e incluso mover la cola de plástico.

Imaginemos que queremos enseñar a nuestro perro robot la habilidad básica del fútbol: que cuando tenga el balón delante, ande hacia él y le dé una patada. La estrategia tradicional de la IA consistiría en programar el

robot con las siguientes reglas: da un paso hacia el balón; repítelo hasta que una de las patas toque el balón; entonces da una patada al balón con esa pata. Por supuesto, las descripciones abreviadas como «da un paso hacia el balón», «hasta que una de las patas toque el balón» y «da una patada al balón» deben traducirse cuidadosamente en el funcionamiento detallado de los sensores y motores integrados en el Aibo.



Un perro robot Sony Aibo, a punto de dar una patada a un balón de fútbol robótico.

Estas reglas explícitas pueden bastar para una tarea tan sencilla como esta. Pero cuanto más «inteligente» queramos que sea nuestro robot, más difícil será concretar manualmente las reglas de conducta. Y, por supuesto, es imposible diseñar un conjunto de reglas válido para todas las situaciones. ¿Y si hay un gran charco entre el robot y el balón? ¿Y si hay un cono que impide ver al robot? ¿Y si hay una piedra que no deja que se mueva el balón? Como siempre, el mundo real está atiborrado de casos extremos difíciles de predecir. Lo que promete el aprendizaje por refuerzo es que el

agente —en este caso, nuestro perro robot— puede aprender estrategias flexibles por sí solo, únicamente a través de hacer cosas y de recibir de vez en cuando recompensas (es decir, refuerzo), sin que los humanos tengan que escribir ninguna regla manualmente ni enseñar directamente al agente todas las circunstancias posibles.

Llamaremos a nuestro perro robot Rosie, en recuerdo de mi robot favorito de la televisión, la irónica criada robótica de la clásica serie de dibujos animados *The Jetsons* (*Los supersónicos*).[177] Para facilitar las cosas, supongamos que Rosie viene programada de fábrica con la siguiente habilidad: si hay un balón de fútbol en su campo visual, es capaz de calcular el número de pasos que tendría que dar para llegar hasta él. Este número se llama «estado». En general, el estado de un agente en un momento dado es la percepción que este tiene de su situación actual. Rosie es el más simple de los agentes posibles, en el sentido de que su estado es un único número. Cuando digo que Rosie está «en» un determinado estado X, quiero decir que en ese momento calcula que está a X pasos del balón.

Además de ser capaz de identificar su estado, Rosie tiene integradas tres acciones que puede llevar a cabo: dar un paso adelante, dar un paso atrás y dar una patada. (Si Rosie se sale del límite, está programada para retroceder de inmediato). Como corresponde al condicionamiento instrumental, vamos a dar a Rosie una recompensa solo cuando consiga dar una patada al balón. Que conste que Rosie no sabe de antemano qué estados o acciones merecen la recompensa, ni siquiera si hay recompensa.

Dado que Rosie es un robot, su «recompensa» no es más que un número, por ejemplo diez, que se añade a su «memoria de recompensa». Podemos considerar que el número diez es el equivalente robótico de una golosina para perros. O quizá no. A diferencia de un perro de verdad, Rosie no tiene un deseo intrínseco de golosinas, de números positivos ni de nada. Como detallaré más adelante, en el aprendizaje por refuerzo, un algoritmo creado por humanos guía el proceso de aprendizaje de Rosie en respuesta a las

recompensas; es decir, el algoritmo le dice a Rosie cómo debe aprender de sus experiencias.

El aprendizaje por refuerzo consiste en hacer que Rosie lleve a cabo acciones en una serie de episodios de aprendizaje, cada uno de los cuales comprende cierto número de iteraciones. En cada iteración, Rosie determina su estado actual y elige una acción. Si Rosie recibe una recompensa, entonces aprende algo, como ilustro más adelante. Aquí dejaré que cada episodio dure hasta que Rosie consiga dar una patada al balón, momento en el que recibe la recompensa. Puede ser un proceso muy largo. Como con el adiestramiento de un perro de verdad, hay que tener paciencia.

La figura 23 ilustra un episodio hipotético de aprendizaje. El episodio comienza con el entrenador (yo) colocando a Rosie y el balón en determinadas posiciones iniciales en el campo, con Rosie de frente al balón (figura 23A). Rosie determina su estado actual: a doce pasos del balón. Como Rosie no ha aprendido nada todavía, es una inocente *tabula rasa* y no sabe qué acción debe preferir, así que elige una al azar entre sus tres posibilidades: adelante, atrás, patada. Supongamos que elige dar un paso atrás. Nosotros, los humanos, podemos ver que es una acción equivocada, pero estamos dejando que Rosie descubra por sí sola cómo ejecutar esta tarea.

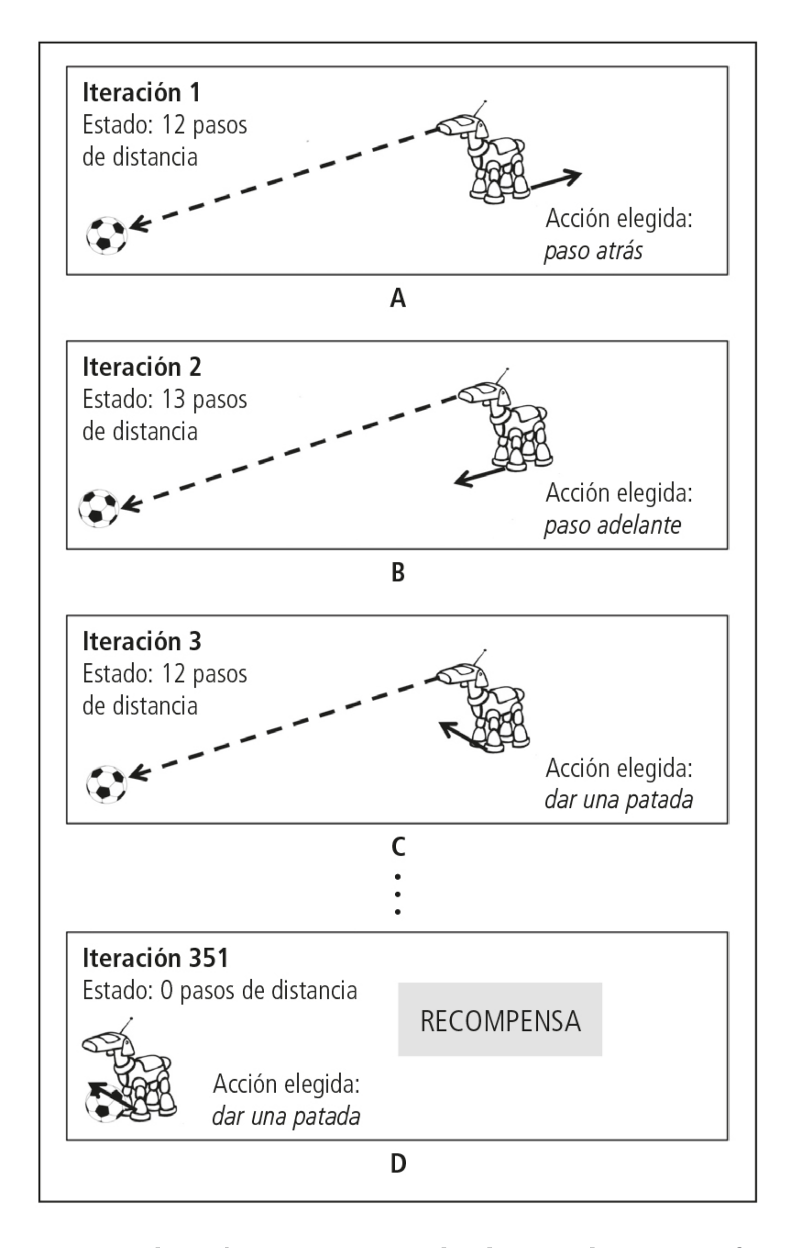


Figura 23. Un hipotético primer episodio de aprendizaje por refuerzo.

En la segunda iteración (figura 23B), Rosie determina su nuevo estado: a trece pasos de distancia del balón. Después elige una nueva acción, otra vez de forma aleatoria: dar un paso adelante. En la tercera iteración (figura 23C), Rosie determina su «nuevo» estado: a doce pasos del balón. Ha vuelto donde empezó, pero Rosie ni siquiera sabe que ya ha estado antes en ese estado. En la forma más pura de aprendizaje por refuerzo, el agente que

está aprendiendo no recuerda sus estados anteriores. En general, recordar estados anteriores utiliza mucha memoria y no es necesario.

En la iteración 3, Rosie —otra vez al azar— elige la acción dar una patada, pero, como está dando la patada en el aire, no obtiene recompensa. Todavía le falta aprender que la patada solo proporciona una recompensa si está junto al balón.

Rosie sigue eligiendo acciones aleatorias, sin saber ninguna valoración, durante muchas iteraciones. Pero en algún momento, por ejemplo en la iteración número 351, Rosie se encuentra por pura casualidad junto al balón y elige dar una patada (figura 23D). Por fin obtiene una recompensa y la utiliza para aprender algo.

¿Qué aprende Rosie? Aquí adoptamos la estrategia más sencilla de aprendizaje por refuerzo: al recibir una recompensa, Rosie solo aprende algo sobre el estado y la acción inmediatamente anteriores a la recompensa. En concreto, Rosie aprende que si se encuentra en ese estado (por ejemplo, a cero pasos del balón), llevar a cabo esa acción (por ejemplo, dar una patada) es una buena idea. Pero no aprende nada más. No aprende, por ejemplo, que si está a cero pasos del balón, dar un paso atrás sería una mala decisión. Al fin y al cabo, todavía no lo ha intentado. Por lo que sabe, a lo mejor dar un paso atrás en ese estado le proporcionaría una recompensa mucho mayor. Rosie tampoco aprende en este momento que si está a un paso, ir adelante sería una buena opción. Para eso tiene que esperar al próximo episodio. Aprender demasiado de una sola vez puede ser perjudicial; si a Rosie se le ocurre dar una patada al aire a dos pasos del balón, no queremos que aprenda que esta patada inútil, en realidad, es un paso necesario para obtener la recompensa. En los seres humanos, ese tipo de comportamiento podría considerarse una superstición, es decir, creer sin razón que una acción concreta puede ayudar a provocar un resultado bueno o malo. En el aprendizaje por refuerzo, la superstición es algo que hay que evitar cuidadosamente.

Una noción crucial en el aprendizaje por refuerzo es la del valor de llevar a cabo una acción concreta en un estado determinado. El valor de la acción A en el estado E es un número que refleja la predicción actual del agente sobre cuánta recompensa acabará obteniendo si estando en el estado E, lleva a cabo la acción A y luego sigue ejecutando acciones de gran valor. Me explico. Si su estado actual es «con un bombón en la mano», una acción de gran valor sería llevarse la mano a la boca. Las siguientes acciones de gran valor serían abrir la boca, introducir el chocolate y masticarlo. La recompensa es la deliciosa sensación de comerse el bombón. Llevarse la mano a la boca no proporciona inmediatamente esta recompensa, pero es una acción que va bien encaminada, y si esa persona ha comido chocolate antes, puede predecir la intensidad de la recompensa que le espera. El objetivo del aprendizaje por refuerzo es que el agente aprenda valores que le ayuden a predecir las recompensas que se avecinan (suponiendo que el agente siga haciendo lo que debe después de la acción en cuestión).[178] Como veremos, para aprender los valores que tienen acciones concretas en un estado determinado suelen hacer falta muchos pasos de ensayo y error.

Tabla Q:

Estado	0 pasos de distancia		1 paso de distancia		. . .	10 pasos de distancia		. . .
Acción	*Paso adelante*	0	*Paso adelante*	0	. . .	*Paso adelante*	0	. . .
	Paso atrás	0	*Paso atrás*	0		*Paso atrás*	0	
	Patada	**10**	*Patada*	0		*Patada*	0	

Figura 24. La tabla Q de Rosie tras su primer episodio de aprendizaje por refuerzo.

Rosie registra los valores de sus acciones en una gran tabla en la memoria de su ordenador. Esta tabla, que aparece en la figura 24, enumera todos los estados posibles de Rosie (es decir, todas las distancias posibles a las que podría estar del balón, hasta el límite del campo), y para cada

estado, las posibles acciones. Dado un estado, cada acción que pueda emprender en ese estado tiene un valor numérico; esos valores cambiarán —y serán predicciones más exactas de las futuras recompensas— a medida que Rosie siga aprendiendo. Esta tabla de estados, acciones y valores se denomina tabla Q. Y esta forma de aprendizaje por refuerzo se denomina a veces *Q-learning* (aprendizaje Q). Se utiliza la letra Q (de *quality*) porque la letra V (de valor) se empleó para otra cosa en el artículo original sobre el aprendizaje Q.[179]

Al comenzar el entrenamiento de Rosie, inicio la tabla Q poniendo todos los valores a cero, para empezar con «página en blanco». Cuando Rosie recibe una recompensa por dar una patada al balón al final del episodio 1, el valor de la acción «dar una patada» cuando está en el estado «a cero pasos» se actualiza a diez, el valor de la recompensa. En el futuro, cuando Rosie esté en el estado «a cero pasos», podrá mirar la tabla Q, ver que «dar una patada» tiene el valor más alto —es decir, predice la recompensa más alta— y elegir esa acción en lugar de elegir al azar. Eso es lo que significa «aprender» aquí.

El episodio 1 terminó con Rosie dando una patada al balón. Ahora pasamos al episodio 2 (figura 25), que comienza con Rosie y la pelota en nuevas posiciones (figura 25A). Igual que antes, en cada iteración Rosie determina su estado actual —al principio, a seis pasos de distancia— y elige una acción, para lo que mira su tabla Q. Ahora bien, a estas alturas, los valores de las acciones en su estado actual siguen siendo todos cero; todavía no hay ninguna información que le ayude a elegir entre ellas. Así que Rosie vuelve a elegir una acción de forma aleatoria: dar un paso atrás. Y vuelve a elegir atrás en la siguiente iteración (figura 25B). Al entrenamiento de nuestro perro robot le queda mucho por delante.

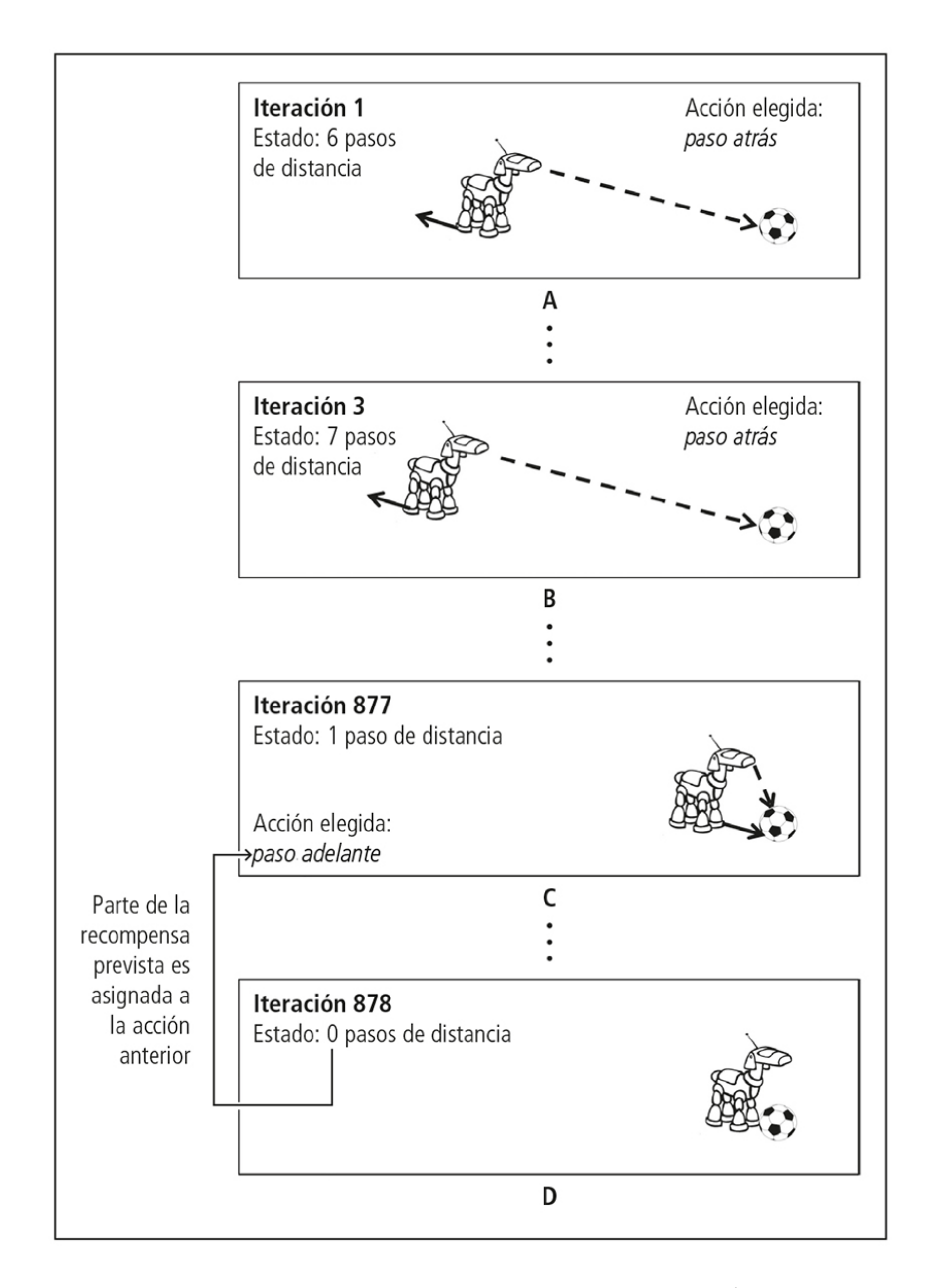


Figura 25. Segundo episodio de aprendizaje por refuerzo.

Todo continúa como antes, hasta que las pruebas de ensayo y error al azar de Rosie la llevan a un paso de la pelota (figura 25C) y elige dar un paso adelante. De repente Rosie se encuentra con la pata junto a la pelota (figura 25D) y ve que la tabla Q tiene información sobre este estado. En concreto, dice que su estado actual —a cero pasos del balón— tiene una acción —dar una patada— que, según se predice, desemboca en una recompensa de diez.

Ahora puede utilizar esta información aprendida en el episodio anterior para elegir qué acción ejecuta, que es dar una patada. Pero aquí está la esencia del aprendizaje Q: ahora Rosie puede aprender algo sobre la acción (dar un paso adelante) que llevó a cabo en el estado inmediatamente anterior (a un paso de distancia). Eso es lo que hizo que ahora esté en la excelente posición en la que se encuentra. El valor de la acción «dar un paso adelante» en el estado «a un paso de distancia» se actualiza en la tabla Q para darle un valor más alto, una parte del valor de la acción «dar una patada a cero pasos», que proporciona directamente una recompensa. Aquí he actualizado este valor a ocho (figura 26).

Tabla Q:

Estado	**0 pasos de distancia**		**1 paso de distancia**		. . .	**10 pasos de distancia**		. . .
Acción	*Paso adelante*	0	***Paso adelante***	**8**	. . .	*Paso adelante*	0	. . .
	Paso atrás	0	*Paso atrás*	0		*Paso atrás*	0	
	Patada	**10**	*Patada*	0		*Patada*	0	

Figura 26. La tabla Q de Rosie después del segundo episodio de aprendizaje por refuerzo.

Ahora la tabla Q le dice a Rosie que está muy bien dar una patada cuando se encuentra en el estado «a cero pasos» y que está casi igual de bien dar un paso adelante cuando se encuentra en el estado «a un paso». La próxima vez que Rosie se encuentre en el estado «a un paso de distancia», tendrá algo de información sobre qué acción debe ejecutar y la capacidad de adquirir una actualización para la acción inmediatamente anterior: dar un paso adelante en el estado «a dos pasos». Es importante que los valores de las acciones aprendidas se reduzcan («se descuenten») a medida que están más alejadas en el tiempo respecto a la recompensa real, para que el sistema aprenda una forma eficaz de llegar hasta ella.

El aprendizaje por refuerzo —en este caso, la actualización gradual de los valores de la tabla Q— continúa, episodio tras episodio, hasta que Rosie aprende por fin a ejecutar su tarea desde cualquier punto de partida inicial. El algoritmo de aprendizaje Q es una forma de asignar valores a las acciones en un estado determinado, incluidas las acciones de su segundo episodio de aprendizaje por refuerzo, que no proporcionan directamente recompensas, pero que preparan el terreno para los estados relativamente escasos en los que el agente sí recibe recompensas.

Escribí un programa que simulaba el proceso de aprendizaje Q de Rosie que acabo de describir. Al principio de cada episodio, Rosie se colocaba frente al balón, a un número aleatorio de pasos de distancia (con un máximo de veinticinco y un mínimo de cero). Como ya he mencionado, si Rosie se salía de los límites, mi programa simplemente la hacía volver a entrar. Cada episodio terminaba cuando Rosie conseguía alcanzar el balón y darle una patada. Comprobé que Rosie tardaba unos trescientos episodios en aprender a ejecutar esta tarea a la perfección, independientemente de dónde empezara.

Este ejemplo de «entrenamiento de Rosie» plasma en gran parte la esencia del aprendizaje por refuerzo, pero he dejado fuera muchas cuestiones que afrontan los investigadores del aprendizaje por refuerzo cuando se trata de tareas más complejas.[180] Por ejemplo, en las tareas del mundo real, la percepción que tiene el agente de su estado suele ser incierta, a diferencia de Rosie, que sabe perfectamente cuántos pasos de distancia la separan del balón. Un robot que juegue al fútbol en la realidad solo podría calcular aproximadamente la distancia, o incluso no sabría con certeza qué objeto pequeño y de color claro del campo de fútbol es exactamente el balón. Tampoco hay seguridad sobre las consecuencias de una acción: por ejemplo, la acción de dar un paso adelante puede hacer que el robot avance diferentes distancias según el terreno, o incluso hacer que se caiga o choque

con un obstáculo invisible. ¿Cómo puede lidiar el aprendizaje por refuerzo con este tipo de incertidumbres?

Además, ¿cómo debe elegir el agente que está aprendiendo cada acción en cada paso? Una estrategia ingenua sería elegir siempre la acción que en la tabla Q tenga el valor más alto para el estado actual. Pero esta estrategia tiene un inconveniente: es posible que otras acciones aún no probadas proporcionen una recompensa mayor. ¿Con qué frecuencia hay que investigar, llevar a cabo acciones que todavía no se han probado, y con qué frecuencia se deben elegir acciones que ya se prevé que van a proporcionar alguna recompensa? Cuando vamos a un restaurante, ¿pedimos siempre platos que ya hemos probado y nos han parecido buenos, o probamos algo nuevo porque quizá el menú contenga opciones incluso mejores? La decisión de hasta qué punto hay que probar nuevas acciones y hasta qué punto explotar (es decir, aferrarse a) las acciones comprobadas se llama equilibrio entre exploración y explotación. Y conseguir el equilibrio adecuado es fundamental para que el aprendizaje por refuerzo tenga éxito.

Estos son ejemplos de temas investigados en la actualidad por la creciente comunidad de personas que trabajan en el aprendizaje por refuerzo. Al igual que en el campo del aprendizaje profundo, el diseño de sistemas de aprendizaje por refuerzo sigue siendo un trabajo difícil (y a veces lucrativo), dominado por un grupo relativamente pequeño de expertos que, como sus homólogos del aprendizaje profundo, dedican mucho tiempo a ajustar los hiperparámetros. (¿Cuántos episodios de aprendizaje deben permitirse? ¿Cuántas iteraciones por episodio deben permitirse? ¿Cuánto debería «descontarse» una recompensa a medida que retrocede en el tiempo?, y así sucesivamente).

Obstáculos en el mundo real

Vamos a dejar a un lado estas cuestiones por ahora para fijarnos en dos grandes obstáculos que pueden surgir al extrapolar nuestro ejemplo de

«entrenar a Rosie» al aprendizaje por refuerzo en tareas del mundo real. En primer lugar, está la tabla Q. En tareas complejas del mundo real —pensemos, por ejemplo, en un coche robótico que aprende a conducir en una ciudad llena de gente— es imposible definir un pequeño conjunto de «estados» que puedan enumerarse en una tabla. Un estado concreto de un coche en un momento dado sería algo así como la totalidad de los datos de sus cámaras y otros sensores. Eso significa que un coche autónomo, en la práctica, se enfrenta a un número infinito de estados posibles. Aprender mediante una tabla Q como la del ejemplo de Rosie es imposible. Por eso, la mayoría de los métodos modernos de aprendizaje por refuerzo utilizan una red neuronal en lugar de una tabla Q. El trabajo de la red neuronal consiste en aprender qué valores deben asignarse a las acciones en un estado determinado. En concreto, la entrada que recibe la red es el estado actual, y las salidas que emite son las estimaciones del valor de todas las acciones posibles que el agente puede llevar a cabo en ese estado. Lo que se espera es que la red pueda aprender a agrupar estados relacionados en conceptos generales («Se puede seguir hacia delante con seguridad» o «Para inmediatamente para no chocar con un obstáculo»).

El segundo escollo es la dificultad de llevar a cabo el proceso de aprendizaje durante muchos episodios en el mundo real con un robot de verdad. Ni siquiera nuestro ejemplo de Rosie es factible. Imaginemos a una persona iniciando un nuevo episodio —saliendo al campo para preparar el robot y el balón— cientos de veces, por no hablar de esperar a que el robot ejecute cientos de acciones en cada episodio. No habría tiempo suficiente. Además, se podría correr el riesgo de que el robot se dañara por elegir una acción equivocada, como dar una patada a un muro de hormigón o avanzar hasta tirarse por un precipicio.

Igual que hice con Rosie, los profesionales del aprendizaje por refuerzo suelen abordar este problema construyendo simulaciones de robots y de entornos, y desarrollando todos los episodios de aprendizaje en la

simulación y no en el mundo real. A veces, este método funciona bien. Se ha entrenado a robots mediante simulaciones para que, entre otras tareas, caminen, salten, agarren objetos y conduzcan un coche con control remoto, y los robots han sido capaces, con distintos niveles de éxito, de trasladar las habilidades aprendidas durante la simulación al mundo real.[181] Sin embargo, cuanto más complejo e impredecible es el entorno, menos éxito tienen los intentos de transferir lo aprendido en la simulación al mundo real. Con estas dificultades, es lógico que hasta ahora los mayores éxitos del aprendizaje por refuerzo no se hayan producido en robótica, sino en campos que pueden simularse perfectamente en un ordenador. En concreto, los éxitos más conocidos del aprendizaje por refuerzo están en el ámbito de los juegos. La aplicación del aprendizaje por refuerzo a los juegos es el tema del próximo capítulo.

---

[176] A. Sutherland, «What Shamu Taught Me About a Happy Marriage», *The New York Times*, 25 de junio de 2006, www.nytimes.com/2006/06/25/fashion/whatshamu-taught-me-about-a-happy-marriage.html.

[177] thejetsons.wikia.com/wiki/Rosey.

[178] Para ser más exactos, este método del aprendizaje por refuerzo, denominado aprendizaje de valores, no es el único posible. Un segundo método, denominado aprendizaje de políticas, tiene como objetivo aprender directamente qué acción llevar a cabo en un estado determinado, en lugar de aprender primero los valores numéricos de las acciones.

[179] C. J. Watkins y P. Dayan, «Q-Learning», *Machine Learning* 8, n.os 3-4 (1992), pp. 279-292.

[180] Para una introducción detallada y técnica al aprendizaje por refuerzo véase R. S. Sutton y A. G. Barto, *Reinforcement Learning: An Introduction*, 2.ª ed., Cambridge, Mass.: MIT Press, 2017, incompleteideas.net/book/the-book-2nd.html.

[181] Por ejemplo, véanse los siguientes trabajos: P. Christiano *et al.*, «Transfer from Simulation to Real World Through Learning Deep Inverse Dynamics Model», arXiv:1610.03518 (2016); J. P. Hanna y P. Stone, «Grounded Action Transformation for Robot Learning in Simulation», en *Proceedings of the Conference of the American Association for Artificial Intelligence* (2017), pp. 3834-3840; A. A. Rusu *et al.*, «Sim-to-Real Robot Learning from Pixels with Progressive Nets», en *Proceedings of the First Annual Conference on Robot Learning, CoRL* (2017); S. James, A. J. Davison y E. Johns, «Transferring End-to-End Visuomotor Control from Simulation to Real World for a Multi-stage Task», en *Proceedings of the First Annual Conference on Robot Learning, CoRL*

(2017); M. Cutler, T. J. Walsh y J. P. How, «Real-World Re-inforcement Learning via Multifidelity Simulators», *IEEE Transactions on Robotics* 31, n.º 3 (2015), pp. 655-671.

09

A jugar

Desde los primeros tiempos de la IA, los entusiastas están obsesionados con crear programas capaces de ganar a los humanos en los juegos. A finales de los años cuarenta, Alan Turing y Claude Shannon, dos de los fundadores de la era informática, escribieron programas para jugar al ajedrez incluso antes de que existieran ordenadores capaces de ejecutar su código. En las décadas siguientes, muchos jóvenes fanáticos de los juegos se han sentido impulsados a aprender a programar para conseguir que el ordenador jugara a su juego favorito, ya sean las damas, el ajedrez, el *backgammon*, el go, el póquer o, más recientemente, los videojuegos.

En 2010, un joven científico británico y entusiasta de los juegos llamado Demis Hassabis, junto con dos amigos, puso en marcha en Londres una empresa llamada DeepMind Technologies. Hassabis es una figura pintoresca y compleja dentro del mundo de la IA moderna. Un prodigio del ajedrez que ya ganaba campeonatos a los seis años, empezó a programar videojuegos profesionalmente a los quince y fundó su propia empresa de videojuegos a los veintidós. Además de sus actividades empresariales, hizo un doctorado en Neurociencia Cognitiva en el University College de Londres para avanzar en su propósito de crear una IA inspirada en el cerebro. Hassabis y sus colegas fundaron DeepMind Technologies para «abordar [las] cuestiones realmente fundamentales» de la inteligencia

artificial.[182] Quizá no es extraño que el grupo DeepMind pensara que los videojuegos eran el escenario adecuado para abordar esas cuestiones. Los videojuegos son, en opinión de Hassabis, «una especie de microcosmos del mundo real, solo que […] más limpios y restringidos».[183]

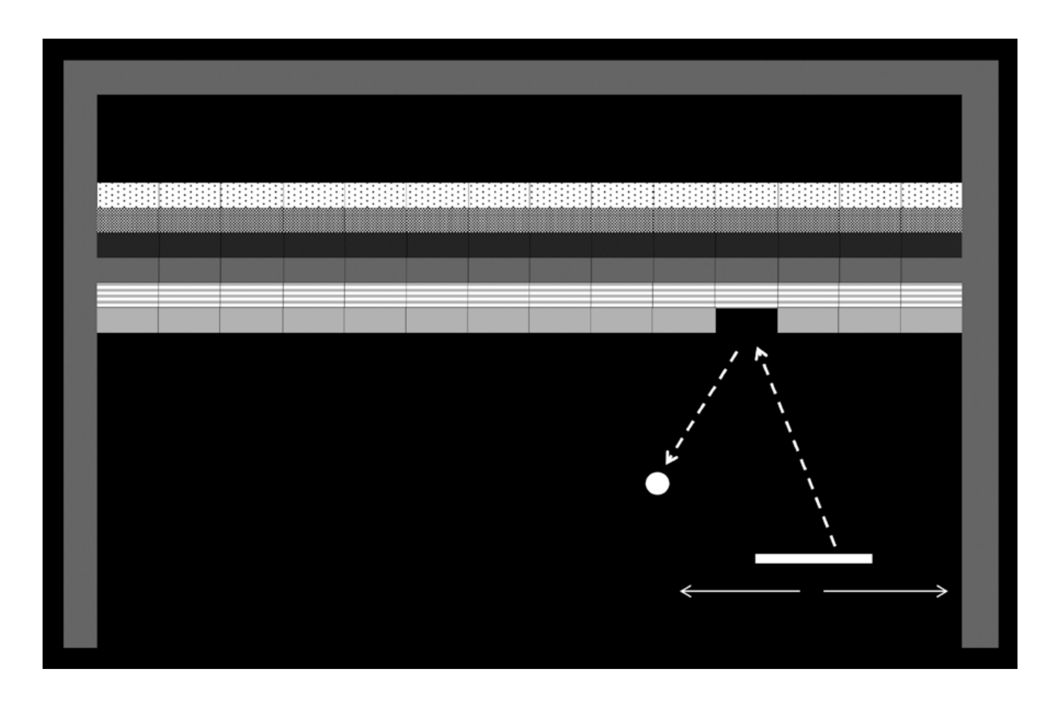


Figura 27. Ilustración del juego Breakout de Atari.

Independientemente de lo que opine cada uno sobre los videojuegos, si a alguien le gusta más lo «limpio y restringido» y menos el «mundo real», quizá le merezca la pena crear programas de IA para jugar a videojuegos de Atari de los años setenta y ochenta. Esto es exactamente lo que decidió hacer el grupo de DeepMind. Según su edad e intereses, puede que recuerden algunos juegos clásicos como *Asteroids*, *Space Invaders*, *Pong* y *Ms. Pac-Man*. ¿Les suena alguno? Con sus gráficos sencillos y controlados con *joystick*, eran unos juegos suficientemente fáciles para que los aprendieran los niños pequeños, pero suficientemente difíciles para mantener el interés de los adultos.

Veamos el juego *Breakout*, para un solo jugador, ilustrado en la figura 27. El jugador utiliza el *joystick* para mover una «pala» (el rectángulo blanco en

la parte inferior derecha) hacia delante y hacia atrás. La pala golpea una «pelota» (el círculo blanco) para dar contra «ladrillos» rectangulares de diferentes colores. La pelota también puede rebotar en las «paredes» grises de los lados. Si la pelota golpea uno de los ladrillos (los rectángulos con dibujos), el ladrillo desaparece, el jugador gana puntos y la pelota rebota. Los ladrillos de las capas superiores valen más puntos que los de las capas inferiores. Si la bola toca el «suelo» (la parte inferior de la pantalla), el jugador pierde una de las cinco vidas, y si todavía le queda alguna, sale una nueva bola. El objetivo del jugador es obtener la máxima puntuación durante las cinco vidas.

Hay una interesante nota al margen. *Breakout* fue el resultado del intento de Atari de crear una versión para un solo jugador de su juego *Pong*, que había tenido gran éxito. El diseño y la implantación de *Breakout* se le encargaron inicialmente, en 1975, a un empleado veinteañero llamado Steve Jobs. Sí, ese Steve Jobs (más tarde, cofundador de Apple). Jobs no tenía suficientes conocimientos de ingeniería para hacer un buen trabajo con *Breakout*, así que reclutó a su amigo Steve Wozniak, de veinticinco años (más tarde, el otro cofundador de Apple), para que le ayudara en el proyecto. Wozniak y Jobs completaron el diseño del soporte físico de *Breakout* en cuatro noches; se ponían a trabajar cada noche cuando Wozniak acababa su jornada en Hewlett-Packard. Cuando salió a la venta, *Breakout* se hizo tan popular como *Pong* entre los jugadores.

Si les invade la nostalgia pero no han conservado su vieja consola Atari 2600, todavía hay muchos sitios web que ofrecen *Breakout* y otros juegos. En 2013, un grupo de investigadores canadienses de IA lanzó una plataforma llamada Arcade Learning Environment que permitía probar sistemas de aprendizaje automático en cuarenta y nueve de estos juegos. [184] Esta fue la plataforma que usó el grupo DeepMind en su trabajo sobre aprendizaje por refuerzo.

Aprendizaje Q profundo

El grupo DeepMind combinó el aprendizaje por refuerzo, en concreto el aprendizaje Q, con redes neuronales profundas para crear un sistema capaz de aprender a jugar a videojuegos de Atari. El grupo denominó a su método aprendizaje Q profundo. Para explicar cómo funciona voy a usar de ejemplo *Breakout*, pero DeepMind utilizó el mismo método con todos los juegos de Atari. Vamos a ponernos un poco técnicos, así que prepárense (o sáltense esta parte).

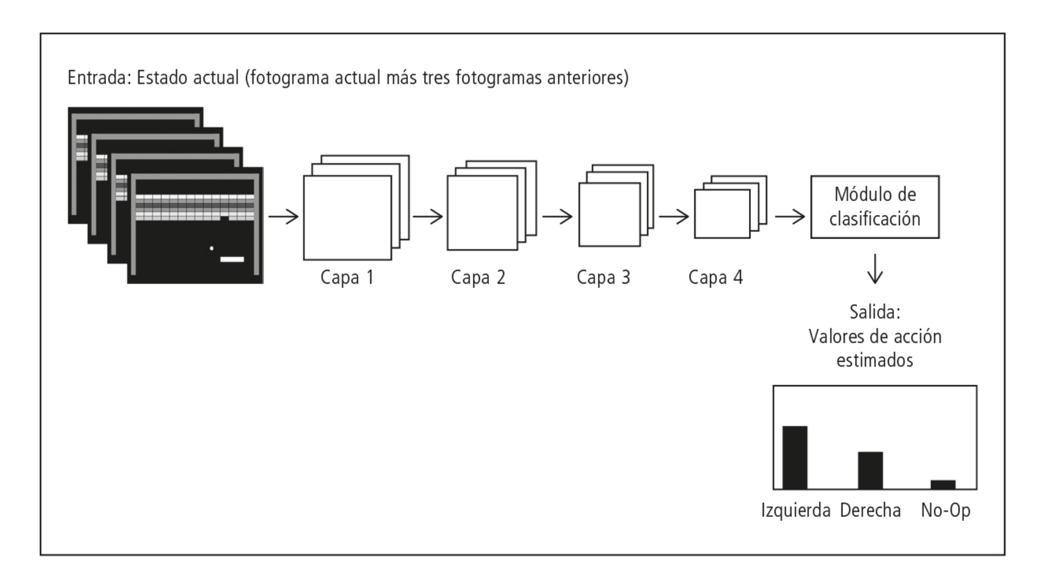


Figura 28. Ilustración de una red Q profunda (DQN) para Breakout.

Recuerden cómo utilizamos el aprendizaje Q para entrenar al perro robot Rosie. En un episodio de aprendizaje Q, en cada iteración, el agente de aprendizaje (Rosie) hace lo siguiente: averigua su estado actual, busca ese estado en la tabla Q, utiliza los valores de la tabla para elegir una acción, lleva a cabo esa acción, quizá recibe una recompensa y —este es el paso de aprendizaje— actualiza los valores de su tabla Q.

El aprendizaje Q profundo de DeepMind es exactamente igual, salvo que el lugar de la tabla Q lo ocupa una red neuronal convolucional. Voy a seguir el ejemplo de DeepMind y a llamar a esta red Red Q profunda (Deep Q-Network, DQN). La figura 28 ilustra una DQN similar (aunque más

sencilla) a la utilizada por DeepMind para aprender a jugar a *Breakout*. La entrada que llega a la DQN es el estado del sistema en un momento dado, que aquí se define como el «fotograma» actual —los píxeles de la pantalla actual— más tres fotogramas anteriores (los píxeles de la pantalla de tres pasos temporales previos). Esta definición de estado proporciona al sistema una pequeña cantidad de memoria, que resulta útil en este caso. Las salidas que emite la red son los valores estimados para cada acción posible, teniendo en cuenta el estado de entrada. Las acciones posibles son las siguientes: mover la pala a la izquierda, mover la pala a la derecha y no-op («no operar», es decir, no mover la pala). La red en sí es una ConvNet prácticamente idéntica a la que describí en el capítulo 4. En lugar de los valores de una tabla Q, como vimos en el ejemplo de Rosie, en el aprendizaje Q profundo lo que se aprende son los pesos de esta red.

El sistema de DeepMind aprende a jugar a *Breakout* a lo largo de muchos episodios. Cada episodio corresponde a una partida del juego y cada iteración durante un episodio corresponde a una única acción del sistema. En concreto, en cada iteración el sistema introduce su estado en la DQN y decide una acción basándose en los valores de salida de la DQN. El sistema no siempre elige la acción con el valor estimado más alto; como he mencionado antes, el aprendizaje por refuerzo exige un equilibrio entre exploración y explotación.[185] El sistema lleva a cabo la acción elegida (por ejemplo, mover la pala un poco hacia la izquierda) y quizá recibe una recompensa si la pelota golpea uno de los ladrillos. A continuación, el sistema lleva a cabo el paso de aprendizaje, es decir, actualiza los pesos en la DQN mediante retropropagación.

¿Cómo se actualizan los pesos? Esto es lo que verdaderamente distingue el aprendizaje supervisado del aprendizaje por refuerzo. Como vimos en capítulos anteriores, la retropropagación funciona cambiando los pesos de una red neuronal para reducir el error en las salidas de la red. Con el aprendizaje supervisado, medir este error es sencillo. ¿Recuerdan nuestra

ConvNet hipotética del capítulo 4, cuyo objetivo era aprender a clasificar fotos como «perro» o «gato»? Si una foto de entrenamiento que constituía la entrada mostraba un perro, pero la confianza con la que se emitía «perro» era solo del 20 por ciento, entonces el error para esa salida sería 100 % − 20 % = 80 %; es decir, en teoría, el valor de salida debería haber sido ochenta puntos más alto. La red podía calcular el error porque tenía una etiqueta que le había proporcionado un ser humano.

Sin embargo, en el aprendizaje por refuerzo no tenemos etiquetas. Un fotograma concreto del juego no está etiquetado con la acción que debe emprenderse. Entonces, ¿cómo asignamos un error a una salida emitida en este caso?

He aquí la respuesta. Recordemos que para el agente de aprendizaje, el valor de una acción en el estado actual es su cálculo de la recompensa que recibirá al final del episodio si elige esa acción (y continúa eligiendo acciones de gran valor). Esta estimación debería ser más acertada cuanto más se acerque el final del episodio, cuando pueda contar las recompensas reales que ha recibido. El truco está en suponer que las salidas de la red en la iteración actual están más cerca de ser correctas que sus salidas en la iteración anterior. Entonces, el aprendizaje consiste en ajustar los pesos de la red (mediante retropropagación) para reducir al mínimo la diferencia entre los resultados de la iteración actual y la anterior. Richard Sutton, uno de los creadores de este método, lo llama «aprender una conjetura a partir de una conjetura».[186] Yo lo llamaría más bien «aprender una conjetura a partir de una conjetura mejor».

En resumen, en vez de aprender a ajustar los valores de salida a las etiquetas proporcionadas por los humanos, la red aprende a hacer que esos valores sean coherentes entre una iteración y la siguiente, basándose en la hipótesis de que las iteraciones posteriores dan mejores estimaciones del valor que las anteriores. Este método de aprendizaje se denomina aprendizaje por diferencia temporal.

Para recapitular, así es como funciona el aprendizaje Q profundo para el juego de *Breakout* (y todos los demás juegos de Atari). El sistema da su estado actual como entrada a la red Q profunda. Esta genera un valor para cada acción posible. El sistema elige y ejecuta una acción, lo que da como resultado un nuevo estado. Ahora lleva a cabo el paso de aprendizaje: el sistema introduce su nuevo estado en la red, que genera un nuevo conjunto de valores para cada acción. La diferencia entre ese conjunto de valores y el anterior se considera el «error» de la red; la retropropagación utiliza ese error para cambiar los pesos de la red. Estos pasos se repiten a lo largo de muchos episodios (las jugadas de la partida). Que quede claro que todo esto —la red Q profunda, el *joystick* virtual y el propio juego— son programas que se ejecutan en un ordenador.

Este es, en definitiva, el algoritmo desarrollado por los investigadores de DeepMind, aunque emplearon algunos trucos para mejorarlo y acelerarlo. [187] Al principio, antes de que la red haya aprendido mucho, sus valores de salida son bastante aleatorios, y la forma de jugar del sistema también parece bastante aleatoria. Pero poco a poco, a medida que la red aprende a ponderar pesos que mejoran las salidas que emite, la capacidad de juego del sistema mejora, en muchos casos de forma espectacular.

El agente de 650 millones de dólares

El grupo DeepMind aplicó su método de aprendizaje Q profundo a los cuarenta y nueve juegos de Atari en el Entorno de Aprendizaje Arcade. Aunque los programadores utilizaron la misma arquitectura de red y los mismos ajustes de hiperparámetros para todos los juegos, el sistema aprendió cada juego desde cero; es decir, los conocimientos del sistema (los pesos de la red) aprendidos para un juego no se transferían cuando el sistema empezaba a aprender a jugar al siguiente. Cada juego exigía un entrenamiento con miles de episodios, pero eso era relativamente rápido gracias a los potentes ordenadores de la empresa.

Después de entrenar una red Q profunda para cada juego, DeepMind comparó el nivel de juego de la máquina con el de un «probador de juegos profesional» humano, al que se le permitieron dos horas de práctica antes de ser evaluado. ¿Parece un trabajo divertido? Solo para quien disfrute cuando lo humilla un ordenador. Los programas de aprendizaje Q profundo de DeepMind demostraron que jugaban mejor que el probador humano en más de la mitad de los juegos. Y en la mitad de esos juegos, los programas consiguieron resultados más de dos veces mejores que el humano. Y en la mitad de esos, los programas fueron más de cinco veces mejores. Un ejemplo especialmente llamativo fue el de *Breakout*, donde el programa DQN obtuvo una puntuación media más de diez veces superior a la del probador humano.

¿Qué aprendieron a hacer exactamente esos programas sobrehumanos? Cuando DeepMind lo indagó, descubrió que sus programas habían descubierto varias estrategias muy inteligentes. Por ejemplo, el programa entrenado con *Breakout* había descubierto un enrevesado truco que muestra la figura 29. Aprendió que si la bola era capaz de derribar ladrillos y formar un túnel estrecho que los atravesara, entonces rebotaba entre el «techo» y la parte superior de la capa de ladrillos y derribaba a toda velocidad los de arriba, muy valiosos, sin que el jugador tuviera que mover la pala en absoluto.

DeepMind presentó este trabajo en 2013, en una conferencia internacional sobre aprendizaje automático.[188] El público quedó deslumbrado. Menos de un año después, Google anunció que había adquirido DeepMind por 440 millones de libras (unos 475 millones de euros de la época), es de suponer que debido a estos resultados. Sí, el aprendizaje por refuerzo a veces proporciona grandes recompensas.

Con todo ese dinero en los bolsillos y los recursos de Google a sus espaldas, DeepMind —ahora llamada Google DeepMind— se propuso un objetivo de más envergadura, que de hecho se había considerado durante

mucho tiempo uno de los «grandes retos» de la IA: crear un programa que aprendiera a jugar al go mejor que cualquier ser humano. El programa AlphaGo de DeepMind se basa en una larga historia de la IA con los juegos de mesa. Empezaré con un breve repaso de esa historia, que nos ayudará a explicar cómo funciona AlphaGo y por qué es tan importante.

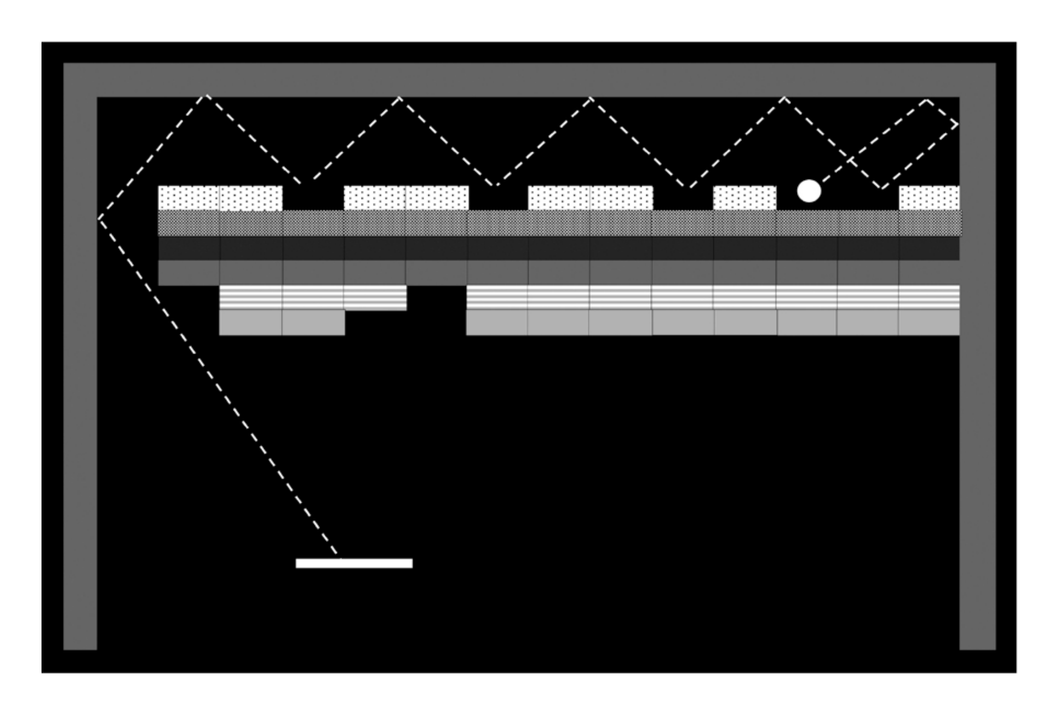


Figura 29. El jugador de Breakout de DeepMind descubrió la estrategia de hacer túneles a través de los ladrillos, lo que le permitió destruir rápidamente los ladrillos superiores, de alto valor, a base de rebotar en el «techo».

Damas y ajedrez

En 1949, el ingeniero Arthur Samuel se incorporó al laboratorio de IBM en Poughkeepsie, Nueva York, e inmediatamente se puso a programar una de las primeras versiones del ordenador 701 de IBM para jugar a las damas. Quien tenga alguna experiencia en programación informática sabrá valorar la dificultad de lo que se proponía: como señala un historiador, «Samuel fue la primera persona que programó en serio el 701, así que el sistema no tenía utilidades dignas de tal nombre [en definitiva, no tenía sistema operativo]. En concreto, no tenía ensamblador y tenía que escribirlo todo utilizando los

códigos de operación y las direcciones».[189] Para que lo entiendan los lectores no programadores, esto es algo así como construir una casa solo con una sierra de mano y un martillo. El programa de Samuel para jugar a las damas fue uno de los primeros programas de aprendizaje automático; es más, fue él quien acuñó el término *aprendizaje automático*.

El jugador de damas de Samuel se basaba en el método de búsqueda en un árbol de juego, que es la base de todos los programas de juegos de mesa hasta hoy (incluido AlphaGo, que describiré más adelante). La figura 30 ilustra parte de un árbol de juego para las damas. La «raíz» del árbol (dibujada por costumbre en la parte superior, a diferencia de la raíz de un árbol natural) muestra el tablero de damas al principio, antes de que ninguno de los jugadores se haya movido. Las «ramas» que salen de la raíz desembocan en todos los movimientos posibles para las damas del primer jugador (aquí, las negras). Hay siete jugadas posibles (para simplificar, la figura no muestra más que tres de ellas). Para cada uno de esos siete movimientos de las negras, hay siete posibles movimientos de respuesta de las blancas (no aparecen todos), y así sucesivamente. Cada uno de los tableros de la figura 30, que muestra una posible disposición de las piezas, se denomina una posición del tablero.

Imaginemos que estamos jugando a las damas. En cada turno, podríamos armar mentalmente una pequeña parte de este árbol. Podríamos decir: «Si hago esta jugada, mi oponente puede hacer esta otra, en cuyo caso yo podría hacer esa otra, lo que me permitiría dar un salto». La mayoría de la gente, incluidos los mejores jugadores, tienen en cuenta solo unos cuantos movimientos posibles y no piensan más que en unos pocos pasos antes de decidir su jugada. Un ordenador rápido, por el contrario, puede hacer ese tipo de predicciones a una escala mucho mayor. ¿Qué le impide al ordenador analizar todos los movimientos posibles y ver qué secuencia lleva con más rapidez a la victoria? El problema es el mismo tipo de incremento exponencial que vimos en el capítulo 3 (el del rey, el sabio y los

granos de arroz). Una partida de damas, por término medio, tiene alrededor de cincuenta jugadas, lo que significa que el árbol de juego de la figura 30 podría extenderse hacia abajo durante cincuenta niveles. En cada nivel, hay una media de seis o siete ramas a partir de cada posición posible del tablero. Esto significa que el número total de posiciones del tablero en el árbol podría ser superior a más de seis elevado a la quincuagésima potencia (6^{50}), un número increíblemente enorme. Un ordenador hipotético que pudiera ver un billón de posiciones de tablero por segundo tardaría más de 10^{19} años en examinar todas las posiciones de tablero de un único árbol de juego. (Como suele hacerse, podemos comparar esta cifra con la edad del universo, que es solo del orden de 10^{10} años). Está claro que una búsqueda completa del árbol de juego no es factible.

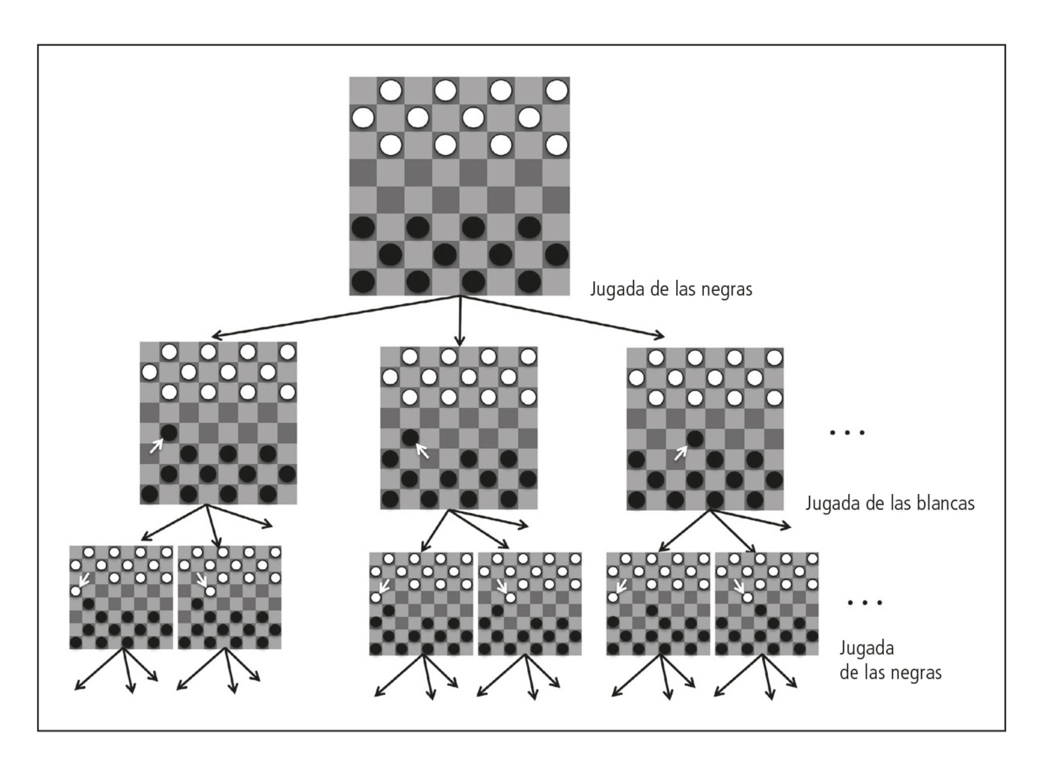


Figura 30. Parte de un árbol de juego para las damas. Para simplificar, esta figura muestra solo tres movimientos posibles desde cada posición del tablero. Las flechas blancas apuntan desde la casilla anterior de una pieza movida hasta su casilla actual.

Por suerte, los ordenadores pueden jugar bien sin hacer una búsqueda tan exhaustiva. En cada uno de sus turnos, el programa de Samuel para jugar a las damas creaba (en la memoria del ordenador) una pequeña parte de un árbol de juego como el de la figura 30. La raíz del árbol era la posición actual del jugador en el tablero, y el programa, con los conocimientos integrados sobre las reglas de las damas, generaba todas las jugadas legales que podía hacer desde esa posición. Luego generaba todas las que el oponente podía hacer desde cada una de las posiciones resultantes, y así sucesivamente hasta cuatro o cinco turnos (o «capas») por delante.[190]

Después, el programa evaluaba las posiciones de tablero que aparecían al final del proceso de anticipación; en la figura 30, serían las posiciones de tablero de la fila inferior del árbol parcial. Evaluar una posición de tablero significa asignarle un valor numérico que calcula la probabilidad de que proporcione una victoria al programa. El programa de Samuel usaba una función de evaluación que daba puntos, treinta y ocho en total, por varios elementos del tablero, como la ventaja de las negras en número total de piezas, el número de reyes de las negras y cuántas damas negras estaban cerca de ser reyes. Estas características específicas las había escogido Samuel basándose en sus conocimientos sobre las damas. Después de evaluar cada una de las posiciones de la fila inferior del tablero, el programa empleaba un algoritmo clásico, llamado minimax, que usaba esos valores —a partir del final del proceso de anticipación— para calificar las posibles jugadas inmediatas del programa desde su posición actual en el tablero. Entonces, el programa elegía la jugada mejor valorada.

Lo que se intuye aquí es que la función de evaluación será más precisa cuando se aplique a las posiciones del tablero con el juego más avanzado; por tanto, la estrategia del programa es primero mirar todas las posibles secuencias de movimientos con unos cuantos pasos de adelanto y luego aplicar la función de evaluación a las posiciones del tablero resultantes. Entonces, el minimax retropropaga las evaluaciones en el árbol, lo que

genera una clasificación de todas las posibles jugadas inmediatas desde la posición actual del tablero.[191]

Lo que aprendía el programa era qué elementos del tablero debían incluirse en la función de evaluación en un turno determinado, así como la forma de ponderar esos distintos elementos al sumar sus puntos. Samuel experimentó con varios métodos de aprendizaje en su sistema. En la versión más interesante, el sistema aprendía mientras jugaba. El método de aprendizaje era un poco complicado y no voy a detallarlo aquí, pero tenía algunos aspectos precursores del actual aprendizaje por refuerzo.[192]

A la hora de la verdad, el jugador de damas de Samuel llegó a tener el nivel nada despreciable de «un jugador por encima de la media», aunque de ninguna manera el de un campeón. Algunos jugadores aficionados dijeron que era «difícil pero derrotable».[193] Pero, sobre todo, el programa fue una bendición publicitaria para IBM: al día siguiente de que Samuel lo enseñara en la televisión nacional, en 1956, el precio de las acciones de IBM subió quince puntos. Esta fue la primera de varias ocasiones en las que IBM vio subir el precio de sus acciones después de una exhibición de algún programa de juego capaz de ganar a los humanos; el ejemplo más reciente es la subida del precio de las acciones que se produjo después de que se emitieran, con enorme éxito de audiencia, una serie de victorias de su programa Watson en el concurso *Jeopardy!*

No obstante, aunque el programa jugador de damas de Samuel fue un hito importante en la historia de la IA, he hecho esta digresión más que nada para introducir tres conceptos muy importantes: el árbol de juego, la función de evaluación y el aprendizaje a base de jugar contra sí mismo.

Deep Blue

Si bien el programa de damas de Samuel, «difícil pero derrotable», era extraordinario, en especial para su época, no refutaba verdaderamente la idea que la gente tenía de sí misma como seres de inteligencia singular.

Aunque una máquina pudiera ganar a campeones humanos de damas (como acabó ocurriendo en 1994),[194] jugar bien a las damas nunca se había considerado una muestra de inteligencia general. El ajedrez era otra cosa. En palabras de Demis Hassabis, de DeepMind: «Durante décadas, los principales informáticos creyeron que, dado el prestigio tradicional del ajedrez como demostración por excelencia del intelecto humano, un ordenador que jugara de forma competente al ajedrez no tardaría en superar también todas las demás aptitudes humanas».[195] Muchas personas, incluidos los pioneros de la IA Allen Newell y Herbert Simon, compartían esta visión glorificada del ajedrez; en 1958, Newell y Simon escribieron: «Si alguien pudiera diseñar una máquina capaz de jugar con éxito al ajedrez, sentiría que ha entrado en el corazón del esfuerzo intelectual humano».[196]

El ajedrez es mucho más complejo que las damas. Por ejemplo, antes he dicho que en las damas hay, por término medio, seis o siete jugadas posibles desde cualquier posición del tablero. En cambio, en el ajedrez hay una media de treinta y cinco jugadas desde cualquier posición del tablero. Eso hace que el árbol de juego del ajedrez sea muchísimo mayor que el de las damas. Durante décadas, los programas de ajedrez han ido mejorando al mismo ritmo que la velocidad de los ordenadores. En 1997, IBM obtuvo su segundo gran triunfo en el mundo de los juegos con Deep Blue, un programa de ajedrez que derrotó al campeón mundial Garry Kasparov en una partida multijuego retransmitida a todo el mundo.

Deep Blue utilizaba casi el mismo método que el programa de damas de Samuel: en un turno dado, creaba un árbol de juego parcial usando la posición actual del tablero como raíz; aplicaba su función de evaluación a la capa más lejana del árbol y luego recurría al algoritmo minimax para retropropagar los valores en el árbol con el fin de decidir qué jugada hacer. Las principales diferencias entre el programa de Samuel y Deep Blue eran que este tenía una mayor capacidad de anticipación en su árbol de juego,

una función de evaluación más compleja (específica del ajedrez), unos conocimientos de ajedrez programados manualmente y equipos paralelos especiales para hacerlo funcionar muy rápido. Además, a diferencia del programa de damas de Samuel, en Deep Blue el aprendizaje automático no era central.

Como había ocurrido con el programa de damas de Samuel, la victoria de Deep Blue contra Kasparov provocó un aumento significativo del precio de las acciones de IBM.[197] Además, la derrota generó una consternación considerable en los medios de comunicación por las connotaciones sobre la inteligencia sobrehumana, así como dudas sobre si la gente seguiría sintiéndose motivada para jugar al ajedrez. Sin embargo, en las décadas transcurridas desde Deep Blue, la humanidad se ha adaptado. Como escribió Claude Shannon en 1950, una máquina capaz de superar a los humanos en ajedrez «nos obligará a admitir la posibilidad del pensamiento automático o a restringir aún más nuestro concepto de pensamiento».[198] Esto es lo que ha sucedido. Ahora se considera que jugar al ajedrez de forma sobrehumana es algo que no requiere inteligencia general. Deep Blue no es inteligente en el sentido que le damos hoy a la palabra. No puede hacer otra cosa que jugar al ajedrez, y no tiene ni idea de lo que significa para los seres humanos «jugar una partida» o «ganar». (Una vez oí decir a un conferenciante: «Puede que Deep Blue venciera a Kasparov, pero no lo disfrutó»). Además, el ajedrez ha sobrevivido —e incluso prosperado— como actividad humana estimulante. Hoy en día, los jugadores humanos utilizan programas informáticos de ajedrez como instrumento para entrenarse, del mismo modo que un jugador de béisbol puede practicar con una máquina lanzadora. ¿Es esto consecuencia de la evolución de nuestra noción de inteligencia, que los avances en IA ayudan a aclarar? ¿O es otro ejemplo de la máxima de John McCarthy: «En cuanto funciona, todo el mundo deja de llamarlo IA»?[199]

El gran reto del go

El go existe desde hace más de dos mil años y está considerado uno de los juegos de mesa más difíciles. Si ustedes no son jugadores de go, no se preocupen; nada de lo que voy a decir aquí exige conocer previamente el juego. Pero es útil saber que se trata de un juego muy prestigioso, sobre todo en el este de Asia, donde es enormemente popular. «El go es un pasatiempo amado por emperadores y generales, intelectuales y niños prodigio», escribe el académico y periodista Alan Levinovitz. Y cita a continuación al campeón surcoreano de go Lee Sedol: «En el mundo occidental existe el ajedrez, pero el go es incomparablemente más sutil e intelectual».[200]

El go es un juego con reglas bastante sencillas, pero que tiene lo que podríamos llamar complejidad emergente. En cada turno, un jugador coloca una pieza de su color (blanco o negro) en un tablero de diecinueve por diecinueve casillas, de acuerdo con las reglas sobre dónde se pueden colocar las piezas propias y cómo capturar las del rival. A diferencia del ajedrez, con su jerarquía de peones, alfiles, reinas y todo lo demás, en el go las piezas («piedras») son todas iguales. Lo que cada jugador tiene que analizar con rapidez para decidir una jugada es la disposición de las piedras en el tablero.

Crear un programa que juegue bien al go ha sido uno de los objetivos de la IA desde los primeros tiempos, pero la complejidad del juego ha dificultado extraordinariamente esta tarea. En 1997, el mismo año en que Deep Blue derrotó a Kasparov, los mejores programas de go aún podían caer derrotados con facilidad por un jugador medio. Deep Blue, recordaremos, era capaz de hacer una gran cantidad de predicciones desde cualquier posición del tablero y luego usar su función de evaluación para asignar valores a futuras posiciones, en las que cada valor predecía si una posición concreta del tablero desembocaría en una victoria. Los programas de go no pueden utilizar esta estrategia por dos motivos. En primer lugar, el

tamaño de un árbol de juego en el go es mucho mayor que en el ajedrez. Mientras que un jugador de ajedrez debe elegir entre una media de treinta y cinco jugadas posibles a partir de una posición dada en el tablero, un jugador de go tiene una media de doscientas cincuenta posibilidades. Incluso con un equipo informático especial, no es factible hacer una búsqueda a las bravas, como las de Deep Blue, en el árbol del go. En segundo lugar, nadie ha conseguido crear una buena función de evaluación para las posiciones del tablero de go. Es decir, nadie ha sido capaz de construir una fórmula verdaderamente capaz de examinar una posición de tablero en el go y predecir quién va a ganar. Los mejores jugadores (humanos) de go confían en su capacidad de reconocer patrones y en una «intuición» difícil de expresar.

Los investigadores de IA no han descubierto todavía cómo codificar la intuición en una función de evaluación. Por eso, en 1997, el mismo año en el que Deep Blue derrotó a Kasparov, el periodista George Johnson escribió en *The New York Times*: «Cuando un ordenador derrote a un campeón humano de go —si alguna vez lo consigue—, será la señal de que la inteligencia artificial está empezando a ser tan buena como la real».[201] Quizá suene familiar; es lo mismo que la gente solía decir sobre el ajedrez. Johnson citaba la predicción de un entusiasta del go: «Puede que pasen cien años hasta que un ordenador gane a los humanos al go; puede que incluso más». Apenas veinte años más tarde, AlphaGo, que había aprendido a jugar al go mediante aprendizaje Q profundo, venció a Lee Sedol en una partida de cinco juegos.

AlphaGo contra Lee Sedol

Antes de explicar cómo funciona AlphaGo, recordemos sus espectaculares victorias contra Lee Sedol, uno de los mejores jugadores de go del mundo. Incluso después de haber visto que AlphaGo había vencido al entonces campeón europeo de go, Fan Hui, medio año antes, Lee seguía confiando en

que iba a ganar: «Creo que [el nivel de AlphaGo] no está a la altura del mío… Por supuesto, habrá habido muchas actualizaciones en los últimos cuatro o cinco meses, pero no le habrá dado tiempo para estar a mi nivel».[202]

Tal vez usted sea una de los más de doscientos millones de personas que vieron por internet algún rato de la partida AlphaGo-Lee en marzo de 2016. Estoy segura de que es el mayor número de espectadores, con diferencia, que ha tenido ninguna partida de go en los veinticinco siglos de historia del juego. Tras la primera partida, es posible que compartiera la reacción de Lee ante su derrota: «Estoy en estado de *shock*, lo reconozco… No creía que AlphaGo fuera a jugar de una forma tan perfecta».[203]

El juego «perfecto» de AlphaGo incluyó muchas jugadas que provocaron sorpresa y admiración entre los comentaristas humanos. Pero, a mitad de la segunda partida, AlphaGo hizo una jugada que dejó patidifusos incluso a los mayores expertos en el go. Como informó *Wired*:

> Al principio, Fan Hui [el campeón europeo de go antes mencionado] pensó que la jugada era bastante extraña. Pero luego le vio la belleza. «No es una jugada humana. Nunca he visto a un humano hacer esta jugada —dice—. Qué preciosidad». Es una palabra que repite una y otra vez. Preciosa. Preciosa. Preciosa… «Es una jugada muy sorprendente», dijo uno de los comentaristas de la partida en inglés, un jugador de go de gran talento. Entonces el otro se rio y dijo: «Pensé que era un error». Pero quizá el más sorprendido fue Lee Sedol, que se levantó y abandonó la sala. «Ha tenido que ir a lavarse la cara o algo así para recuperarse», dijo el primer comentarista.[204]

Sobre esta misma jugada, *The Economist* señaló: «Curiosamente, los maestros humanos del go hacen a veces este tipo de jugadas. En japonés las llaman *kami no itte* (“la mano de Dios” o “jugadas divinas”)».[205]

AlphaGo ganó esa partida y la siguiente. Pero en la cuarta partida, Lee tuvo su propio momento *kami no itte*, que encarna la complejidad del juego y la capacidad de intuición de los mejores jugadores. La jugada de Lee sorprendió a los comentaristas, pero enseguida reconocieron que podía ser letal para su adversario. Un cronista escribió: «Sin embargo, AlphaGo no

parecía darse cuenta de lo que estaba pasando. No se había encontrado con nada similar […] en los millones y millones de partidas que había jugado consigo misma. En la rueda de prensa posterior a la partida, se le preguntó a Sedol en qué había pensado mientras movía su piedra. Dijo que había sido la única jugada que había podido ver».[206]

AlphaGo perdió la cuarta partida, pero remontó para ganar la quinta y, por tanto, la competición. En los medios de comunicación de masas se trató como un nuevo duelo entre Deep Blue y Kasparov, con innumerables artículos de opinión sobre lo que el triunfo de AlphaGo significaba para el futuro de la humanidad. Pero este triunfo era aún más significativo que la victoria de Deep Blue: la IA había superado un escollo todavía mayor que el del ajedrez, y lo había hecho de una forma mucho más impresionante. A diferencia de Deep Blue, AlphaGo había adquirido sus habilidades mediante el aprendizaje por refuerzo jugando contra sí misma.

Demis Hassabis señaló que «lo que distingue a los mejores jugadores de go [es] su intuición», y que «lo que hemos hecho con AlphaGo es introducir, con redes neuronales, este aspecto de la intuición, si se quiere llamar así».[207]

Cómo funciona AlphaGo

Ha habido varias versiones diferentes de AlphaGo, así que, para mantener un orden, DeepMind empezó a llamarlas como los campeones humanos de go a los que habían derrotado —AlphaGo Fan y AlphaGo Lee—, algo que a mí me recordaba a la imagen de los cráneos de los enemigos vencidos en la colección de un vikingo digital. Estoy segura de que no era ese el objetivo de DeepMind. En cualquier caso, tanto AlphaGo Fan como AlphaGo Lee utilizaron una enrevesada mezcla de aprendizaje Q profundo, «búsqueda de árbol de Monte Carlo», aprendizaje supervisado y conocimientos especializados de go. Sin embargo, un año después de la partida contra Lee Sedol, DeepMind desarrolló una versión del programa que era al mismo

tiempo más sencilla y mejor que las versiones anteriores. Esta nueva versión se llama AlphaGo Zero porque, a diferencia de su predecesora, partía con «cero» conocimientos sobre el go, aparte de las reglas.[208] En cien partidas de AlphaGo Lee contra AlphaGo Zero, esta última las ganó todas. Además, DeepMind aplicó los mismos métodos (aunque con redes diferentes y distintas reglas de juego incorporadas) para aprender a jugar al ajedrez y al *shogi* (también conocido como ajedrez japonés).[209] A esta serie de métodos la llamaron AlphaZero. En esta sección describiré cómo funcionaba AlphaGo Zero, pero para no extenderme, llamaré a esta versión sencillamente AlphaGo.

La palabra *intuición* tiene un aura de misterio, pero la intuición de AlphaGo (si queremos llamarla así) nace de su combinación de aprendizaje Q profundo con un método inteligente llamado «búsqueda de árbol de Monte Carlo». Vamos a detenernos a descifrar este largo nombre. Primero, la parte de «Monte Carlo». Montecarlo es, por supuesto, la parte más glamurosa del diminuto Principado de Mónaco, en la Riviera francesa, conocido por sus casinos llenos de famosos y millonarios, sus carreras de coches y su frecuente aparición en las películas de James Bond. Pero en ciencia y matemáticas, «Monte Carlo» es una familia de algoritmos informáticos, el llamado método de Monte Carlo, que se utilizó por primera vez durante el Proyecto Manhattan para ayudar a diseñar la bomba atómica. El nombre procede de la idea de que un ordenador puede utilizar cierto grado de aleatoriedad —como la de la icónica ruleta giratoria del Casino de Montecarlo— para resolver problemas matemáticos difíciles.

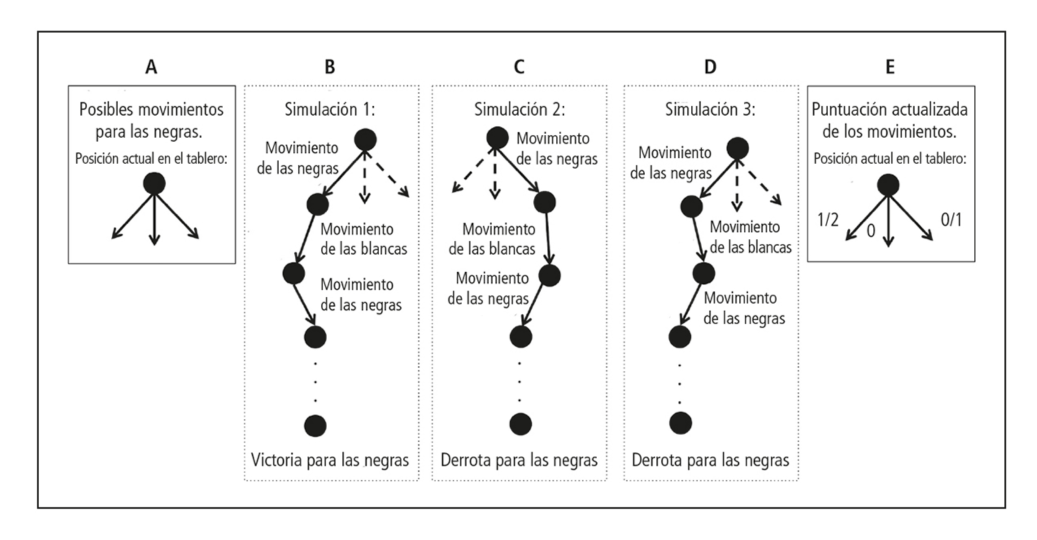


Figura 31. Ilustración de la búsqueda de árbol de Monte Carlo.

La búsqueda de árbol de Monte Carlo es una versión del método de Monte Carlo concebida específicamente para programas informáticos de juego. Igual que la función de evaluación de Deep Blue, la búsqueda de árbol de Monte Carlo se utiliza para asignar una puntuación a cada jugada posible desde una posición determinada del tablero. Sin embargo, como he explicado anteriormente, el uso de una búsqueda muy anticipada en el árbol de juego no es factible en el go, y nadie ha sido capaz de crear una buena función de evaluación para las posiciones del tablero en este juego. La búsqueda de árbol de Monte Carlo funciona de forma diferente.

La figura 31 ilustra la búsqueda de árbol de Monte Carlo. Primero, veamos la figura 31A. El círculo negro representa la posición actual del tablero, es decir, la disposición de las piezas sobre el tablero en el turno actual. Supongamos que nuestro programa de jugar al go está jugando con las negras y es su turno. Supongamos, para simplificar, que las negras tienen tres movimientos posibles, representados por las tres flechas. ¿Cuál debe elegir?

Si el jugador que tiene las negras dispusiera de tiempo suficiente, podría hacer una «búsqueda completa» en el árbol de juego: mirar todas las posibles secuencias de jugadas futuras que se podrían hacer y elegir la que

tenga más probabilidades de llevar a las negras a la victoria. Pero no es posible hacer esta búsqueda exhaustiva; como he explicado antes, ni siquiera todo el tiempo transcurrido desde el comienzo del universo es suficiente para hacer una búsqueda completa de árbol en el go. Con la búsqueda de árbol de Monte Carlo, las negras miran solo una fracción minúscula de las posibles secuencias que podrían derivar de cada jugada, cuentan cuántas victorias y derrotas obtienen esas secuencias hipotéticas y usan esos cálculos para evaluar cada uno de los posibles movimientos. La aleatoriedad inspirada en la rueda de la ruleta se utiliza para decidir cómo hacer la búsqueda por adelantado.

Más en concreto, para elegir una jugada desde su posición actual, las negras «imaginan» (es decir, simulan) varias formas posibles en las que se podría desarrollar la partida, como se ilustra en la figura 31B-D. En cada una de estas simulaciones, las negras comienzan en su posición actual, eligen aleatoriamente una de sus posibles jugadas, después (desde la nueva posición del tablero) eligen aleatoriamente una jugada para su rival (las blancas), y así sucesivamente, hasta que la partida simulada termine con una victoria o una derrota para las negras. Este tipo de simulación, que parte de una posición determinada en el tablero, es lo que se llama un *roll-out* (lanzamiento o despliegue) a partir de esa posición.

En la figura se puede ver que, en los tres lanzamientos, las negras han ganado una vez y han perdido dos. Ahora las negras pueden asignar una puntuación a cada jugada posible desde su posición actual en el tablero (figura 31E). La jugada 1 (flecha a la izquierda) ha participado en dos lanzamientos, uno de los cuales acabó en victoria, por lo que la puntuación es 1 sobre 2. La jugada 3 (flecha a la derecha) ha participado en un lanzamiento que acabó en derrota, por lo que su puntuación es 0 sobre 1. La jugada que representa la flecha central no se intentó, así que su puntuación es de 0. Además, el programa registra estadísticas similares sobre todos los movimientos intermedios de los lanzamientos. Una vez terminada esta

ronda de búsqueda de árbol de Monte Carlo, el programa puede utilizar las puntuaciones actualizadas para decidir cuál de las posibles jugadas parece más prometedora; aquí, la 1. Entonces el programa puede hacerla en la partida real.

Cuando dije antes que durante un lanzamiento el programa elige jugadas para sí mismo y para sus rivales de forma aleatoria, lo que ocurre en realidad es que el programa elige jugadas con criterio probabilístico, basándose en las puntuaciones que tienen esas jugadas de rondas anteriores de búsqueda de árbol de Monte Carlo. Cuando un lanzamiento termina con una victoria o una derrota, el algoritmo actualiza todas las puntuaciones de las jugadas realizadas durante esa partida para reflejar la victoria o la derrota.

Al principio, la elección de jugadas por parte del programa a partir de una posición dada del tablero es muy aleatoria (hace el equivalente a girar una ruleta para escoger un número), pero a medida que el programa hace más lanzamientos y genera más estadísticas, está cada vez más predispuesto a elegir las jugadas que en tiradas anteriores han proporcionado más victorias.

Es decir, el algoritmo de búsqueda de árbol de Monte Carlo no tiene que adivinar a partir de la posición en el tablero qué jugada tiene más probabilidades de desembocar en victoria, sino que utiliza sus lanzamientos para recoger datos sobre cuántas veces una jugada determinada proporciona verdaderamente una victoria o una derrota. Cuantos más lanzamientos haga el algoritmo, mejores serán sus estadísticas. Como antes, el programa necesita encontrar el equilibrio entre la explotación (elegir las jugadas de mayor puntuación durante un lanzamiento) y la exploración (elegir a veces jugadas de menor puntuación sobre las que el programa todavía no tiene grandes estadísticas). En la figura 31 se muestran tres lanzamientos; la búsqueda de árbol de Monte Carlo de AlphaGo hizo casi dos mil lanzamientos por turno.

Los informáticos de DeepMind no inventaron la búsqueda de árbol de Monte Carlo. La primera vez que se propuso fue en el contexto de los árboles de juego en 2006, con el resultado de una enorme mejora de aptitudes en los programas informáticos de go. Pero estos programas seguían sin poder vencer a los mejores jugadores humanos. Uno de los problemas era que para generar estadísticas suficientes a partir de los lanzamientos puede hacer falta mucho tiempo, sobre todo en go, con su inmensa cantidad de jugadas posibles. El grupo DeepMind se dio cuenta de que podía mejorar su sistema si completaba la búsqueda de árbol de Monte Carlo con una red neuronal convolucional profunda. Si se utiliza la posición actual del tablero como información de entrada, AlphaGo utiliza una red neuronal convolucional profunda entrenada para asignar un valor aproximado a todas las jugadas posibles desde la posición actual. Después, la búsqueda de árbol de Monte Carlo usa esos valores para iniciar su búsqueda: en vez de empezar escogiendo las jugadas al azar, utiliza los valores emitidos por la ConvNet como indicador de qué jugadas iniciales son preferibles. Imaginemos que AlphaGo es una persona que mira una posición del tablero: antes de empezar el proceso de Monte Carlo de hacer lanzamientos desde esa posición, la ConvNet le susurra al oído qué posibles jugadas desde la posición actual tienen probabilidades de ser las mejores.

A la inversa, los resultados de la búsqueda de árbol de Monte Carlo sirven para entrenar la ConvNet. Imaginemos a esa persona que es AlphaGo después de una búsqueda de árbol de Monte Carlo. Los resultados de su búsqueda son nuevas probabilidades asignadas a todas las posibles jugadas, basadas en cuántas veces han desembocado esas jugadas en victorias o derrotas durante los lanzamientos realizados. Esas nuevas probabilidades se utilizan para corregir mediante retropropagación los valores de salida de la ConvNet. Después, esa persona y su rival deciden sus jugadas, que producen una nueva posición en el tablero, y el proceso continúa. En teoría, la red neuronal convolucional aprenderá a reconocer patrones, igual que

hacen los maestros de go. Con el tiempo, la ConvNet desempeñará el papel de «intuición» del programa, que mejora todavía más gracias a la búsqueda de árbol de Monte Carlo.

Igual que su antecesor, el programa de damas de Samuel, AlphaGo aprende jugando contra sí mismo durante muchas partidas (alrededor de cinco millones). Durante su entrenamiento, los pesos de la red neuronal convolucional se actualizan después de cada jugada basándose en la diferencia entre los valores de salida de la red y los valores mejorados después de ejecutar la búsqueda de árbol de Monte Carlo. Entonces, cuando llega el momento de que AlphaGo juegue, por ejemplo, contra un humano como Lee Sedol, la ConvNet entrenada se utiliza en cada turno para generar valores que ayuden a iniciar esa búsqueda de árbol de Monte Carlo.

Con su proyecto AlphaGo, DeepMind demostró que uno de los grandes obstáculos históricos de la IA podía superarse mediante una ingeniosa combinación de aprendizaje por refuerzo, redes neuronales convolucionales y búsqueda de árbol de Monte Carlo (y añadiendo a la mezcla potentes equipos informáticos modernos). Como consecuencia, AlphaGo ha alcanzado un merecido lugar en el panteón de la IA. Pero ¿qué será lo próximo? ¿Se generalizará esta potente combinación de métodos más allá del mundo de los juegos? Esta es la pregunta que trataré en el próximo capítulo.

---

[182] Demis Hassabis, citado en P. Iwaniuk, «A Conversation with Demis Hassabis, the Bullfrog AI Prodigy Now Finding Solutions to the World's Big Problems», PCGamesN, consultado el 7 de diciembre de 2018, www.pcgamesn.com/demis-hassabis-interview.

[183] Citado en «From Not Working to Neural Networking», *The Economist*, 25 de junio de 2016.

[184] M. G. Bellemare *et al.*, «The Arcade Learning Environment: An Evaluation Platform for General Agents», *Journal of Artificial Intelligence Research* 47 (2013), pp. 253-279.

[185] Para ser más técnicos, el programa de DeepMind utilizó lo que se llama un método *epsilon-greedy*, o ε-voraz, para elegir una acción en cada paso de tiempo. Con probabilidad épsilon el programa elige una acción al azar; con probabilidad (1 − épsilon) el programa elige la acción con el

valor más alto. Épsilon es un valor comprendido entre cero y uno. Al principio es cercano a uno y se va reduciendo a lo largo de los episodios de entrenamiento.

[186] R. S. Sutton y A. G. Barto, *Reinforcement Learning: An Introduction*, 2.ª ed., Cambridge, Mass.: MIT Press, 2017, p. 124, incompleteideas.net/book/the-book-2nd.html.

[187] Para más detalles, véase V. Mnih *et al.*, «Human-Level Control Through Deep Reinforcement Learning», *Nature* 518, n.º 7540 (2015), p. 529.

[188] V. Mnih *et al.*, «Playing Atari with Deep Reinforcement Learning», *Proceedings of the Neural Information Processing Systems (NIPS) Conference, Deep Learning Workshop* (2013).

[189] «Arthur Samuel», sitio web de History of Computers, history-computer.com/ModernComputer/thinkers/Samuel.html.

[190] El programa de Samuel utilizaba un número variable de hilos en función del movimiento.

[191] El programa de Samuel también utilizaba un método llamado poda alfa-beta en cada turno para determinar los nodos del árbol del juego que no necesitaban ser evaluados. La poda alfa-beta también era una parte esencial del programa de ajedrez Deep Blue de IBM.

[192] Para más detalles, véase A. L. Samuel, «Some Studies in Machine Learning Using the Game of Checkers», *IBM Journal of Research and Development* 3, n.º 3 (1959), pp. 210-229.

[193] *Ibid.*

[194] J. Schaeffer *et al.*, «CHINOOK: The World Man-Machine Checkers Champion», *AI Magazine* 17, n.º 1 (1996), p. 21.

[195] D. Hassabis, «Artificial Intelligence: Chess Match of the Century», *Nature* 544 (2017), pp. 413-414.

[196] A. Newell, J. Calman Shaw y H. A. Simon, «Chess-Playing Programs and the Problem of Complexity», *IBM Journal of Research and Development* 2, n.º 4 (1958), pp. 320-335.

[197] M. Newborn, *Deep Blue: An Artificial Intelligence Milestone*, Nueva York: Springer, 2003, p. 236.

[198] Citado en J. Goldsmith, «The Last Human Chess Master», *Wired*, 1 de febrero de 1995.

[199] Citado en M. Y. Vardi, «Artificial Intelligence: Past and Future»», *Communications of the Association for Computing Machinery* 55, n.º 1 (2012), p. 5.

[200] A. Levinovitz, «The Mystery of Go, the Ancient Game That Computers Still Can't Win», *Wired*, 12 de mayo de 2014.

[201] G. Johnson, «To Test a Powerful Computer, Play an Ancient Game», *The New York Times*, 29 de julio de 1997.

[202] Citado en «S. Korean Go Player Confident of Beating Google's AI», Yonhap News Agency, 23 de febrero de 2016, english.yonhapnews.co.kr/search1/2603000000.html?cid=AEN20160223003651315.

[203] Citado en M. Zastrow, «'I'm in Shock!': How an AI Beat the World's Best Human at Go», *New Scientist*, 9 de marzo de 2016, www.newscientist.com/article/2079871im-in-shock-how-an-ai-beat-the-worlds-best-human-at-go.

[204] C. Metz, «The Sadness and Beauty of Watching Google's AI Play Go», *Wired*, 11 de marzo de 2016, www.wired.com/2016/03/sadness-beauty-watching-googlesai-play-go.

[205] «For Artificial Intelligence to Thrive, It Must Explain Itself», *The Economist*, 15 de febrero de 2018, www.economist.com/news/science-and-technology/21737018if-it-cannot-who-will-trust-it-artificial-intelligence-thrive-it-must.

[206] P. Taylor, «The Concept of 'Cat Face'», *London Review of Books*, 11 de agosto de 2016.

[207] Citado en S. Byford, «DeepMind Founder Demis Hassabis on How AI Will Shape the Future», *Verge*, 10 de marzo de 2016, www.theverge.com/2016/3/10/11192774/demis-hassabis-interview-alphago-google-deepmind-ai.

[208] D. Silver *et al.*, «Mastering the Game of Go Without Human Knowledge», *Nature*, 550 (2017), pp. 354-359.

[209] D. Silver *et al.*, «A General Reinforcement Learning Algorithm That Masters Chess, Shogi, and Go Through Self-Play», *Science* 362, n.º 6419 (2018), pp. 1140-1144.

10

Más allá de los juegos

En la última década, el aprendizaje por refuerzo ha pasado de ser una rama relativamente desconocida de la IA a ser uno de los enfoques más apasionantes (y financiados) de este campo. El resurgimiento del aprendizaje por refuerzo, sobre todo a efectos del público, se debe en gran parte a los proyectos de DeepMind que he descrito en el capítulo anterior. Los resultados de DeepMind en los juegos de Atari y en go son realmente extraordinarios, importantes y dignos de elogio.

Sin embargo, el desarrollo de programas de juego sobrehumanos no es el verdadero objetivo de la mayoría de los investigadores de IA. Vamos a desviarnos un poco para preguntarnos por las consecuencias que tienen estos éxitos en el progreso general de la IA. Demis Hassabis tiene algo que decir al respecto:

> Los juegos no son más que nuestra plataforma de desarrollo. […] Son la forma más rápida de desarrollar estos algoritmos de IA y probarlos, pero en última instancia queremos que se utilicen en problemas del mundo real y tengan una enorme repercusión en ámbitos como la sanidad y la ciencia. Lo importante es que es una IA general que está aprendiendo a hacer cosas [basándose en] su propia experiencia y sus propios datos.[210]

Vamos a profundizar un poco en ello. ¿Hasta qué punto es general esta IA? ¿Hasta qué punto se puede utilizar en el mundo real, más allá de los juegos?

¿Hasta qué punto aprenden estos sistemas «por sí solos»? ¿Y qué es exactamente lo que aprenden?

Generalidad y «aprendizaje por transferencia»

Mientras navegaba en busca de artículos sobre AlphaGo, internet me ofreció este llamativo titular: «AlphaGo de DeepMind aprendió ajedrez en su tiempo libre».[211] Esta frase es errónea y engañosa, y es importante entender por qué. AlphaGo (en todas sus versiones) solo puede jugar al go. Incluso en el caso de la versión más general, AlphaZero, no se trata de un único sistema que ha aprendido a jugar al go, al ajedrez y al *shogi*. Cada juego tiene su propia red neuronal convolucional que hay que entrenar desde cero para ese juego concreto. A diferencia de los humanos, ninguno de estos programas puede «transferir» nada de lo que ha aprendido sobre un juego para aprender otro diferente.

Lo mismo ocurre con los distintos programas de juego de Atari: cada uno aprende desde cero los pesos de su propia red. Es como si aprendiéramos a jugar a *Pong*, pero luego para aprender a jugar a *Breakout*, tuviéramos que olvidar por completo todo lo que hemos aprendido jugando a *Pong* y empezar de cero.

Una expresión optimista que se usa en el mundo del aprendizaje automático es «aprendizaje por transferencia», que se refiere a la capacidad de un programa de transferir lo que ha aprendido sobre una tarea para que le ayude a hacer otra distinta. En los seres humanos, el aprendizaje por transferencia es automático. Después de aprender a jugar al *ping-pong*, pude transferir algunas de esas habilidades para aprender a jugar al tenis y al bádminton. Saber jugar a las damas me ayudó a aprender a jugar al ajedrez. Cuando era pequeña, tardé un tiempo en aprender a girar el pomo de la puerta de mi habitación, pero, una vez adquirida esa habilidad, pude generalizarla rápidamente a casi cualquier tipo de pomo.

Los humanos hacemos este tipo de transferencia de una tarea a otra de forma aparentemente fácil; nuestra capacidad para generalizar lo que aprendemos es una parte esencial de lo que para nosotros significa pensar. Por tanto, en lenguaje humano, podríamos decir que un sinónimo de «aprendizaje por transferencia» es sencillamente «aprender».

Al contrario de lo que ocurre con los humanos, en la IA actual, la mayor parte del «aprendizaje» no se puede transferir entre tareas relacionadas. En este sentido, estamos todavía lejos de lo que Hassabis llama «IA general». Aunque el tema del aprendizaje por transferencia es uno de los campos de investigación más activos para los profesionales del aprendizaje automático, los avances en este frente son todavía incipientes.[212]

«Sin ejemplos ni orientación de humanos»

A diferencia del aprendizaje supervisado, el aprendizaje por refuerzo promete programas capaces de aprender de verdad por sí solos, simplemente llevando a cabo acciones en su «entorno» y observando el resultado. La afirmación más importante de DeepMind sobre sus resultados, especialmente con AlphaGo, es que su trabajo ha hecho realidad esa promesa: «Nuestros resultados demuestran sin lugar a dudas que la estrategia pura del aprendizaje por refuerzo es totalmente viable, incluso en los terrenos más difíciles: es posible entrenar a un nivel sobrehumano, sin ejemplos ni orientación de humanos y sin ningún conocimiento del terreno aparte de las reglas básicas».[213]

Ya tenemos la afirmación. Ahora veamos las reservas. AlphaGo (o más exactamente, la versión AlphaGo Zero) no utilizó ningún ejemplo humano en su aprendizaje, pero lo de la «orientación» humana es otra historia. Hubo unos cuantos aspectos de orientación humana que fueron fundamentales para su éxito: la arquitectura específica de su red neuronal convolucional, el uso de la búsqueda de árbol de Monte Carlo y la configuración de los numerosos hiperparámetros que estas dos cosas implican. Como ha

señalado el psicólogo e investigador de IA Gary Marcus, AlphaGo «no aprendió» ninguno de estos aspectos cruciales «de los datos, mediante puro aprendizaje de refuerzo. Más bien, los programadores de DeepMind los integraron de forma innata».[214] En realidad, los programas de DeepMind para los juegos de Atari eran mejores ejemplos de «aprendizaje sin orientación humana» que AlphaGo, porque a ellos no se les proporcionaban las reglas del juego (por ejemplo, que el objetivo en *Breakout* es destruir ladrillos) ni tampoco una idea de los «objetos» relevantes para el juego (por ejemplo, la pala o la pelota), sino que aprendían exclusivamente de los píxeles de la pantalla.

Los terrenos más complicados

Es necesario explorar otro aspecto de la afirmación de DeepMind: «incluso en los terrenos más difíciles». ¿Cómo podemos evaluar lo difícil que es un terreno para la IA? Como hemos visto, muchas cosas que los humanos consideramos bastante fáciles (por ejemplo, describir el contenido de una foto) son muy complicadas para los ordenadores. Y, al contrario, muchas cosas que a los humanos nos parecerían horrorosamente difíciles (por ejemplo, multiplicar correctamente dos números de cincuenta cifras) las hacen los ordenadores en una fracción de segundo con un programa de una sola línea.

Una forma de evaluar lo difícil que es un terreno para los ordenadores es ver lo bien que se desenvuelven en él algoritmos muy sencillos. En 2018, un grupo de investigadores de Uber AI Labs descubrió que algunos algoritmos relativamente sencillos eran casi equiparables (y a veces superiores) al método de aprendizaje Q profundo de DeepMind en varios videojuegos de Atari. El algoritmo que más sorprendió por sus buenos resultados fue el de «búsqueda aleatoria»: en lugar de entrenar una red Q profunda mediante aprendizaje por refuerzo durante muchos episodios, es posible probar muchas redes neuronales convolucionales diferentes con

pesos elegidos al azar.[215] Es decir, no hay ningún tipo de aprendizaje, excepto a través de un sistema aleatorio de prueba y error.

Podría parecer que una red con pesos aleatorios tendría unos resultados horribles en un videojuego de Atari. De hecho, la mayoría de estas redes son pésimas jugadoras. Pero los investigadores de Uber siguieron probando nuevas redes con pesos aleatorios y, por fin (en menos tiempo del que se tarda en entrenar una red Q profunda), encontraron unas redes que funcionaban casi tan bien o incluso mejor que las redes entrenadas mediante aprendizaje Q profundo en cinco de los trece juegos que probaron. Otro algoritmo relativamente sencillo, llamado algoritmo genético,[216] superó al aprendizaje Q profundo en siete de trece juegos. Es difícil decir algo de estos resultados, salvo que quizá el ámbito de los juegos de Atari no es tan difícil para la IA como se pensaba en un principio.

No he sabido de nadie que haya intentado una búsqueda aleatoria similar de pesos de red para go. Me sorprendería mucho que funcionara. Dada la larga historia de los intentos de crear programas informáticos para jugar al go, estoy convencida de que el go es un terreno auténticamente difícil para la IA. Sin embargo, como señaló Gary Marcus, los seres humanos juegan a muchos juegos que son aún más difíciles que el go para la IA. Un ejemplo notable que da Marcus es el de las charadas,[217] que, si lo pensamos bien, exige una compleja comprensión visual, lingüística y social que supera con mucho las capacidades de cualquier sistema de IA actual. Si pudiéramos construir un robot capaz de jugar a las charadas tan bien como, por ejemplo, un niño de seis años, creo que entonces podríamos afirmar sin temor a equivocarnos que hemos conquistado varios de los «terrenos más difíciles» para la IA.

¿Qué aprendieron estos sistemas?

Como sucede con otras aplicaciones del aprendizaje profundo, es difícil interpretar lo que las redes neuronales utilizadas en estos sistemas de juego

han aprendido verdaderamente. Al leer las secciones anteriores, quizá hayan notado cómo se deslizaba cierto antropomorfismo sutil en mis descripciones; por ejemplo, he dicho: «El jugador de *Breakout* de DeepMind descubrió la estrategia de construir un túnel a través de los ladrillos».

Es peligrosamente fácil, para mí y para cualquiera, caer en este tipo de lenguaje al hablar del comportamiento de los sistemas de IA. Sin embargo, incluye muchas veces suposiciones inconscientes que pueden no valer para estos programas. ¿El programa de *Breakout* de DeepMind descubrió verdaderamente el concepto de túnel? Gary Marcus nos recuerda que debemos tener cuidado con estas cosas:

> El sistema no ha aprendido nada de eso; no entiende realmente qué es un túnel ni qué es un muro; no ha aprendido más que contingencias concretas para situaciones concretas. Las pruebas de transferencia —en las que el sistema de aprendizaje por refuerzo profundo se encuentra con situaciones ligeramente distintas a las que utilizó para entrenarse— muestran que las soluciones del aprendizaje por refuerzo profundo son con frecuencia muy superficiales.[218]

Marcus se está refiriendo a varios estudios que intentaron averiguar hasta qué punto los sistemas de aprendizaje Q profundo pueden transferir lo aprendido, incluso a versiones con mínimas variaciones del mismo juego. Por ejemplo, un grupo de investigadores estudió un sistema similar al programa de *Breakout* de DeepMind. Descubrieron que, cuando el programa se ha entrenado hasta un nivel «sobrehumano», si la posición de la pala en la pantalla se desplaza unos pocos píxeles hacia arriba, los resultados del sistema empeoran de golpe.[219] Esto hace pensar que el sistema ni siquiera ha aprendido el concepto básico de pala. Otro grupo demostró que, con un sistema de aprendizaje Q profundo entrenado en el juego *Pong*, si se cambia el color de fondo de la pantalla, el rendimiento del sistema disminuye de forma sustancial.[220] Además, en cada caso, el sistema necesita muchos episodios de reentrenamiento para adaptarse a la variación.

Estos son solo dos ejemplos de la incapacidad del aprendizaje Q profundo para generalizar, que contrasta de forma llamativa con la inteligencia humana. No conozco ningún estudio que haya analizado el concepto de túnel en el programa de *Breakout* de DeepMind, pero me atrevo a suponer que el sistema no podría generalizar, por ejemplo, hacer un túnel hacia abajo o hacia los lados sin un reentrenamiento considerable. Como señala Marcus, aunque los humanos atribuimos al programa cierta comprensión de conceptos que consideramos básicos (por ejemplo, *pared*, *techo*, *paleta*, *pelota*, *túnel*), el programa, en realidad, no los entiende:

> Estas demostraciones dejan claro que es equívoco atribuir al aprendizaje por refuerzo profundo la inducción de conceptos como *pared* o *pala*; esas observaciones son lo que la psicología comparativa (animal) llama a veces sobreatribuciones. No es que el sistema Atari aprendiera realmente un concepto sólido de *pared*, sino que el sistema se aproximaba de forma superficial a atravesar paredes en unas circunstancias restringidas y muy entrenadas.[221]

Del mismo modo, aunque AlphaGo exhibiera una «intuición» milagrosa al jugar al go, el sistema no tiene ningún mecanismo, que yo sepa, que le permita generalizar su capacidad para jugar al go —ni siquiera, por ejemplo, a un tablero de go más pequeño o de forma diferente— sin reestructurar y volver a entrenar su red Q profunda.

En definitiva, aunque estos sistemas profundos de aprendizaje Q han logrado un rendimiento sobrehumano en algunas áreas concretas, e incluso muestran algo parecido a la «intuición» en esos terrenos, lo que no poseen es una cosa absolutamente fundamental para la inteligencia humana. Ya se lo llame abstracción, generalización a otros terrenos o aprendizaje por transferencia, conseguir que los sistemas tengan esa capacidad sigue siendo uno de los principales problemas pendientes de la IA.

Hay otro motivo para sospechar que estos sistemas no están aprendiendo conceptos similares a los humanos ni entendiendo sus ámbitos de la misma forma que lo hacen los humanos: igual que los sistemas de aprendizaje supervisado, estos sistemas de aprendizaje Q profundos son vulnerables a ejemplos antagónicos como los que describí en el capítulo 6. Por ejemplo,

un grupo de investigadores ha demostrado que es posible hacer cambios específicos muy pequeños en los píxeles de la entrada de un programa de juego de Atari, unos cambios imperceptibles para los humanos pero que disminuyen considerablemente la capacidad del programa para jugar.

¿Cómo de inteligente es AlphaGo?

Hay algo que debemos tener en cuenta cuando hablamos de juegos como el ajedrez y el go y su relación con la inteligencia humana. Pensemos en los motivos por los que muchos padres animan a sus hijos a apuntarse al club de ajedrez del colegio (o en algunos lugares al club de go) y prefieren verlos jugando al ajedrez (o al go) que sentados en casa viendo la televisión o jugando a videojuegos (lo siento, Atari). Es porque creen que los juegos como el ajedrez o el go enseñan a los niños (o a cualquiera) a pensar mejor: a pensar con lógica, a hacer razonamientos abstractos y a planificar de forma estratégica. Unas aptitudes que les ayudarán durante el resto de sus vidas, unas habilidades generales que cada persona podrá utilizar en todas sus actividades.

En cambio, AlphaGo, a pesar de los millones de partidas que ha jugado durante su entrenamiento, no ha aprendido a «pensar» mejor sobre nada salvo sobre el juego del go. De hecho, no tiene capacidad para reflexionar sobre nada, para razonar sobre nada, para hacer planes sobre nada, salvo el go. Que yo sepa, ninguna de las habilidades que ha aprendido tiene nada de general; ninguna puede transferirse a ninguna otra tarea. AlphaGo es la perfecta representación del síndrome del sabio.

Desde luego, el método de aprendizaje Q profundo utilizado en AlphaGo puede servir para aprender otras tareas, pero el sistema tendría que volver a hacer todo el entrenamiento; tendría que empezar básicamente desde cero para aprender una nueva habilidad.

Esto nos lleva de nuevo a la paradoja de la IA de que «lo fácil es difícil». AlphaGo fue un gran triunfo para la IA; después de aprender en gran parte a

base de jugar contra sí mismo, consiguió derrotar de forma contundente a uno de los mejores jugadores humanos del mundo en un juego que se considera un modelo de destreza intelectual. Pero AlphaGo no muestra una inteligencia de nivel humano tal como se suele definir, o incluso podría decirse que no muestra ningún tipo de inteligencia verdadera. Para los seres humanos, una parte crucial de la inteligencia es, más que poder aprender una habilidad concreta, ser capaces de aprender a pensar y después aplicar de forma flexible nuestro pensamiento a cualquier situación o escollo que nos encontremos. Esta es la verdadera aptitud que queremos que aprendan nuestros hijos cuando juegan al ajedrez o al go. Puede sonar extraño, pero, en este sentido, el niño más modesto del club de ajedrez del colegio es más inteligente que AlphaGo.

De los juegos al mundo real

Por último, consideremos la afirmación de Demis Hassabis de que el objetivo verdaderamente importante de estas demostraciones con juegos es «que se utilicen en problemas del mundo real y tengan una enorme repercusión en ámbitos como la sanidad y la ciencia». En mi opinión, es muy posible que el trabajo de DeepMind sobre el aprendizaje por refuerzo acabe teniendo esa repercusión que busca Hassabis. Pero de los juegos al mundo real hay mucho que recorrer.

Un obstáculo es la necesidad de transferir el aprendizaje. Pero hay otros motivos por los que será difícil extender el éxito del aprendizaje por refuerzo en los juegos al mundo real. Los juegos como *Breakout* y el go son perfectos para el aprendizaje por refuerzo porque tienen unas reglas claras, unas funciones de recompensa sencillas (por ejemplo, recompensas por puntos ganados o por ganar) y relativamente pocas acciones (jugadas) posibles. Además, los jugadores tienen acceso a la «información perfecta»: tienen a la vista todos los elementos del juego en todo momento; no hay partes del «estado» de un jugador ocultas o inciertas.

El mundo real no tiene unos límites tan claros. Douglas Hofstadter ha señalado que la propia noción de un «estado» claramente definido no es nada realista. «Si vemos las situaciones que se dan en el mundo, no están enmarcadas como una partida de ajedrez o de go… Una situación en el mundo no tiene ningún límite; no se sabe qué hay dentro ni qué hay fuera de la situación».[222]

Por poner un ejemplo, imaginemos el uso del aprendizaje por refuerzo para enseñar a un robot a hacer una tarea muy útil en el mundo real: coger los platos sucios amontonados en el fregadero y meterlos en el lavavajillas. (Cuánta armonía traería un robot así a mi familia). ¿Cómo definir el «estado» del robot? ¿Habría que incluir todo lo que está en su campo visual? ¿El contenido del fregadero? ¿El contenido del lavavajillas? ¿Y el perro, que se acerca a lamer los platos y hay que decirle que se vaya? Independientemente de cómo defina su estado, el robot tendría que ser capaz de identificar distintos objetos: por ejemplo, un plato (que debe ir en la bandeja inferior del lavavajillas), una taza de café (que debe ir en la bandeja superior) o una esponja (que no hay que poner en el lavavajillas). Como hemos visto, el reconocimiento de objetos por ordenador todavía está muy lejos de ser perfecto. Además, el robot tendría que razonar sobre objetos que no puede ver, como quizá unas ollas y sartenes escondidas en el fondo del fregadero. También tendría que aprender a coger distintos objetos y colocarlos (con cuidado) cada uno en su sitio. Y para todo eso tendría que aprender a elegir entre una multitud de acciones posibles relacionadas con la colocación del cuerpo del robot, sus «dedos» para agarrar, los motores para controlar el movimiento de los objetos desde el fregadero hasta la ranura correcta del lavavajillas, y así sucesivamente.[223]

Los agentes jugadores de DeepMind necesitaban millones de iteraciones de entrenamiento. Si no queremos millones de platos rotos, tendríamos que entrenar a nuestro robot con una simulación. Los juegos se pueden simular en un ordenador con gran rapidez y precisión; no hay piezas reales que se

muevan ni pelotas que reboten en palas ni ladrillos que exploten. Pero una simulación en la que un robot meta platos en un lavavajillas no es tan fácil. Cuanto más realista es la simulación, más despacio la ejecuta el ordenador, e incluso con un ordenador muy rápido, es enormemente difícil incorporar exactamente a la simulación todas las fuerzas físicas y otros aspectos de la carga de platos. Y no hay que olvidarse del inoportuno perro y otros aspectos impredecibles del mundo real. ¿Cómo podemos saber qué hay que incluir en la simulación y qué se puede dejar fuera?

Todos estos problemas llevaron a Andrej Karpathy, director de IA de Tesla, a señalar que para tareas del mundo real como esta, «básicamente se quiebran todos y cada uno de los supuestos que cumple el go y que aprovecha AlphaGo, y cualquier enfoque que pretendiera dar fruto tendría que ser muy diferente».[224]

Nadie sabe cuál sería ese enfoque. En realidad, el campo del aprendizaje profundo por refuerzo es todavía bastante joven. Los resultados que he descrito en este capítulo pueden considerarse un principio de prueba: la combinación de las redes profundas y el aprendizaje Q funciona sorprendentemente bien en algunos terrenos muy interesantes, aunque limitados, y aunque mi exposición ha puesto de manifiesto algunas de las limitaciones actuales del campo, muchas personas están trabajando en ampliar el aprendizaje por refuerzo para aplicarlo de forma más general. Los programas de juego de DeepMind, en especial, han suscitado enorme interés y entusiasmo en este campo; de hecho, la revista *Technology Review* del MIT dijo que el aprendizaje profundo por refuerzo era una de «las diez tecnologías revolucionarias» de 2017. En los próximos años, a medida que se perfeccione el aprendizaje por refuerzo, esperaré con impaciencia a un robot que llene el lavavajillas y aprenda por sí solo, y quizá que juegue al fútbol y al go en su tiempo libre.

---

[210] Citado en P. Iwaniuk, «A Conversation with Demis Hassabis, the Bullfrog AI Prodigy Now Finding Solutions to the World's Big Problems», *PCGamesN*, consultado el 7 de diciembre de 2018, www.pcgamesn.com/demis-hassabis-interview.

[211] E. David, «DeepMind's AlphaGo Mastered Chess in Its Spare Time», Silicon Angle, 6 de diciembre de 2017, siliconangle.com/blog/2017/12/06/deepmindsalphago-mastered-chess-spare-time.

[212] Como ejemplo, todavía en el ámbito de los juegos, DeepMind publicó un documento en 2018 que describía un sistema de aprendizaje de refuerzo que, según la empresa, mostraba cierto grado de aprendizaje por transferencia en su capacidad para jugar a diferentes juegos de Atari. L. Espeholt *et al.*, «Impala: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures», en *Proceedings of the International Conference on Machine Learning* (2018), pp. 1407-1416.

[213] D. Silver *et al.*, «Mastering the Game of Go Without Human Knowledge», *Nature* 550 (2017), pp. pp. 354-359.

[214] G. Marcus, «Innateness, AlphaZero, and Artificial Intelligence», arXiv:1801. 05667 (2018).

[215] F. P. Such *et al.*, «Deep Neuroevolution: Genetic Algorithms Are a Competitive Alternative for Training Deep Neural Networks for Reinforcement Learning», *Proceedings of the Neural Information Processing Systems (NIPS) Conference, Deep Reinforcement Learning Workshop* (2018).

[216] M. Mitchell, *An Introduction to Genetic Algorithms*, Cambridge, Mass.: MIT Press, 1996.

[217] Marcus, «Innateness, AlphaZero, and Artificial Intelligence».

[218] G. Marcus, «Deep Learning: A Critical Appraisal», arXiv:1801.00631 (2018).

[219] K. Kansky *et al.*, «Schema Networks: Zero-Shot Transfer with a Generative Causal Model of Intuitive Physics», en *Proceedings of the International Conference on Machine Learning* (2017), pp. 1809-1818.

[220] A. A. Rusu *et al.*, «Progressive Neural Networks», arXiv:1606.04671 (2016).

[221] Marcus, «Deep Learning».

[222] Citado en N. Sonnad y D. Gershgorn, «Q&A: Douglas Hofstadter on Why AI Is Far from Intelligent»", *Quartz*, 10 de octubre de 2017, qz.com/1088714/qa-douglashofstadter-on-why-ai-is-far-from-intelligent.

[223] Debo señalar que algunos grupos de robótica han desarrollado robots que colocan los platos en el lavavajillas, aunque ninguno de ellos ha sido entrenado mediante aprendizaje por refuerzo ni ningún otro método de aprendizaje automático, que yo sepa. Estos robots vienen acompañados de algunos vídeos impresionantes (por ejemplo, «Robotic Dog Does Dishes, Plays Fetch», NBC New York, 23 de junio de 2016, www.nbcnewyork.com/news/local/Boston-Dynamics-Dog-Does-Dishes-Brings-Sodas-384140021.html), pero está claro que siguen siendo bastante limitados y que aún no están preparados para resolver las discusiones nocturnas sobre el lavavajillas en mi casa.

[224] A. Karpathy, «AlphaGo, in Context», *Medium*, 31 de mayo de 2017, medium.com/@karpathy/alphago-in-context-c47718cb95a5.

PARTE IV

LA INTELIGENCIA ARTIFICIAL ENTRA EN CONTACTO CON EL LENGUAJE NATURAL

11

Dime con quién andas y te diré qué palabra eres

Les voy a contar una anécdota.

El restaurante

> Un hombre entró en un restaurante y pidió una hamburguesa poco hecha. Cuando esta llegó a la mesa, estaba completamente quemada. La camarera se acercó. «¿Está bien la carne?», le preguntó. «Está estupenda», dijo el hombre, mientras empujaba la silla hacia atrás y se iba del restaurante hecho una furia y sin pagar. La camarera le gritó: «Eh, ¿y la cuenta?». Después se encogió de hombros y murmuró: «¿Por qué se ha puesto así?».[225]

Y ahora déjenme que les pregunte: ¿se comió la hamburguesa?

Imagino que estarán bastante seguros de su respuesta, aunque la historia no explique directamente la cuestión. Es fácil, al menos para los seres humanos, leer entre líneas. Al fin y al cabo, comprender el lenguaje —incluidas las partes que quedan implícitas— es una parte fundamental de la inteligencia humana. No es casualidad que Alan Turing planteara su famoso «juego de imitación» como un concurso de generación y comprensión del lenguaje.

Esta parte del libro trata del procesamiento del lenguaje natural, que significa «hacer que los ordenadores manejen el lenguaje humano». (En el lenguaje de la IA, *natural* significa «humano»). El procesamiento del lenguaje natural (PLN) incluye temas como el reconocimiento del habla, la

búsqueda en internet, la respuesta automática a preguntas y la traducción automática. Igual que hemos visto en capítulos anteriores, el aprendizaje profundo ha sido el motor de la mayoría de los avances recientes en PLN. Describiré algunos de estos avances, y voy a usar la anécdota del «restaurante» para ilustrar algunos de los principales obstáculos que deben superar las máquinas a la hora de utilizar y comprender el lenguaje humano.

La sutileza del lenguaje

Supongamos que queremos crear un programa capaz de leer un fragmento y responder preguntas sobre él. Los sistemas de preguntas y respuestas son un tema central de la investigación actual en PLN, porque la gente quiere utilizar el lenguaje natural para interactuar con los ordenadores (por ejemplo, con Siri, Alexa, Google Now y otros «asistentes virtuales»). Sin embargo, para responder a preguntas sobre un texto como el de la historia del «restaurante», un programa necesitaría habilidades lingüísticas más complejas y un buen conocimiento de cómo funciona el mundo.

¿Se comió el hombre la hamburguesa? Para responder a esta pregunta con certeza, un hipotético programa debería saber que las hamburguesas pertenecen a la categoría «alimentos» y que los alimentos se pueden comer. Debería saber que entrar en un restaurante y pedir una hamburguesa significa que uno piensa comérsela. Y que, en un restaurante, cuando llega lo que hemos pedido, se puede comer. El programa tendría que saber que cuando una persona pide una hamburguesa «poco hecha», en general no quiere comérsela si está «completamente quemada». Debería saber que cuando el hombre dice «Está buenísima», está siendo sarcástico, y que «está» se refiere a la «carne», que es otra forma de referirse a la «hamburguesa». El programa tendría que suponer que si uno sale «hecho una furia» de un restaurante sin pagar, es probable que no se haya comido lo que le han servido.

Es abrumador pensar en todos los conocimientos previos que necesitaría el programa para responder con seguridad a preguntas básicas sobre la anécdota. ¿Dejó el hombre propina a la camarera? El programa tendría que conocer la costumbre de dejar propina en los restaurantes para recompensar un buen servicio. ¿Por qué dijo la camarera: «¿Y la cuenta?»? El programa tiene que averiguar que «cuenta», en este caso, no es, por ejemplo, una cuenta de collar ni una cuenta bancaria, sino lo que tiene que pagar el hombre por la comida. ¿Sabía la camarera que el hombre estaba enfadado? El programa tiene que saber que sí por la pregunta «¿Por qué se ha puesto así?». «Se» se refiere al hombre, y «se ha puesto así» es un modismo que significa «se ha enfadado tanto». ¿Sabía la camarera por qué se fue el hombre del restaurante? Sería útil que nuestro programa supiera que el gesto de «encogerse de hombros» indica que la camarera no entendió por qué se había marchado hecho una furia.

Pensar en lo que necesitaría saber nuestro hipotético programa me recuerda a tratar de responder a las interminables preguntas que me hacían mis hijos cuando eran muy pequeños. Una vez, cuando mi hijo tenía cuatro años, lo llevé conmigo al banco. Me preguntó sencillamente: «¿Qué es un banco?». La respuesta provocó una cascada aparentemente infinita de preguntas de «por qué». «¿Por qué usa dinero la gente?». «¿Por qué la gente quiere tener mucho dinero?». «¿Por qué la gente no puede guardar todo su dinero en casa?». «¿Por qué no puedo fabricar mi propio dinero?». Todas ellas buenas preguntas, pero difíciles de responder sin tener que explicar todo tipo de cosas que quedan fuera de la experiencia de un niño de cuatro años.

En el caso de los ordenadores, la situación es mucho peor. Un niño que escucha la historia del «restaurante» ya posee algunos conceptos sólidos, como *persona*, *mesa* y *hamburguesa*. Los niños tienen un sentido común básico y saben, por ejemplo, que cuando el hombre sale del restaurante ya no está dentro de él, pero las mesas y las sillas probablemente siguen allí. O

que cuando la hamburguesa «llegó», alguien la llevó a su mesa (no llegó sola). Las máquinas actuales no dominan los conceptos detallados e interrelacionados, y carecen de los conocimientos de sentido común que hasta un niño de cuatro años aporta a la comprensión del lenguaje.

No es de extrañar, por tanto, que utilizar y comprender el lenguaje natural sea uno de los retos más difíciles de la IA. El lenguaje es intrínsecamente ambiguo, depende en gran medida del contexto y presupone una gran cantidad de conocimientos previos que los interlocutores tienen en común. Al igual que en otras áreas de la IA, las primeras décadas de investigación sobre PLN se dedicaron a métodos basados en reglas simbólicas, es decir, programas que recibían reglas gramaticales y lingüísticas y las aplicaban a las frases de entrada. Esos métodos no dieron grandes frutos; parece imposible captar las sutilezas del lenguaje aplicando un conjunto de reglas explícitas. En los años noventa, los métodos de PLN basados en reglas se vieron eclipsados por otros estadísticos que conseguían mejores resultados, en los que se empleaban inmensos conjuntos de datos para entrenar algoritmos de aprendizaje automático. En tiempos más recientes, este método basado en datos estadísticos se ha centrado en el aprendizaje profundo. ¿Puede el aprendizaje profundo, unido a los macrodatos, producir ordenadores capaces de manejar el lenguaje humano de forma flexible y fiable?

El reconocimiento de voz y el último 10 por ciento

El reconocimiento automático del habla —la tarea de convertir el lenguaje hablado en texto sobre la marcha— fue el primer gran éxito del aprendizaje profundo en PLN, y me atrevería a decir que es el mayor éxito de la IA hasta hoy en cualquier ámbito. En 2012, al mismo tiempo que el aprendizaje profundo revolucionaba la visión por ordenador, unos grupos de investigación de la Universidad de Toronto, Microsoft, Google e IBM

publicaron un histórico artículo sobre el reconocimiento del habla.[226] Estos grupos habían estado desarrollando redes neuronales profundas para diversos aspectos del reconocimiento del habla: reconocimiento de fonemas a partir de señales acústicas, predicción de palabras a partir de combinaciones de fonemas, predicción de frases a partir de combinaciones de palabras, y así sucesivamente. Según un experto de Google en reconocimiento del habla, el uso de redes profundas supuso «la principal mejora en veinte años de investigación sobre el habla».[227] Ese mismo año, se puso a disposición de los poseedores de móviles con sistema Android un nuevo sistema de reconocimiento de voz mediante redes profundas; dos años más tarde se puso en marcha para el iPhone de Apple, y un ingeniero de la compañía comentó: «Fue una de esas cosas en las que la mejora [de las prestaciones] era tan sustancial que uno lo vuelve a comprobar para asegurarse de que no hay alguien a quien se le ha olvidado un decimal».[228]

Quien haya utilizado algún tipo de tecnología de reconocimiento de voz antes y después de 2012 también habrá notado la gran mejora. El reconocimiento de voz, que hasta 2012 era entre terriblemente frustrante y moderadamente útil, de repente se volvió casi perfecto en algunas circunstancias. Hoy puedo dictar todos mis mensajes de texto y correos electrónicos en la aplicación de reconocimiento de voz de mi teléfono; hace unos momentos, le he leído la historia del «restaurante», hablando a velocidad normal, y ha transcrito correctamente todas las palabras.

Lo que me resulta asombroso es que los sistemas de reconocimiento de voz hagan todo eso sin comprender el significado de lo que están transcribiendo. El sistema de reconocimiento de voz de mi teléfono puede transcribir cada palabra de mi anécdota del restaurante, pero les garantizo que no entiende nada de ella, ni de ninguna otra cosa. Muchos expertos en inteligencia artificial, entre los que me incluyo, habíamos creído que el reconocimiento de voz nunca alcanzaría este nivel sin comprender el lenguaje. Pero se ha visto que estábamos equivocados.

Dicho esto, el reconocimiento automatizado del habla aún no está al «nivel humano», en contra de lo que afirman algunos medios de comunicación. El ruido de fondo puede reducir considerablemente la precisión de estos sistemas; son mucho menos eficaces dentro de un coche en marcha que en una habitación en silencio. Además, en ocasiones, las palabras o frases poco habituales los desconciertan, lo que pone en evidencia que no comprenden lo que están transcribiendo. Por ejemplo, si digo «la *mousse* es mi postre favorito», mi teléfono (Android) lo transcribe «el mus es mi postre favorito». Digo: «El hombre con la cabeza despejada necesitaba un sombrero» y el teléfono lo transcribe como «El hombre con la cabeza despellejada necesitaba un sombrero». No es difícil encontrar frases que confundan a un sistema de reconocimiento del habla. Sin embargo, con el habla cotidiana en un entorno tranquilo, yo diría que la precisión de estos sistemas —medida por el número de palabras correctamente transcritas— es probablemente de entre el 90 y el 95 por ciento de la precisión humana. [229] Si se añaden ruido u otras complicaciones, el acierto disminuye de forma considerable.

Hay una famosa regla general en cualquier proyecto complejo de ingeniería: el primer 90 por ciento del proyecto ocupa el 10 por ciento del tiempo, y el último 10 por ciento ocupa el 90 por ciento del tiempo. Creo que esta regla, en una u otra versión, vale para muchos ámbitos de la IA (los coches autónomos, por ejemplo), y acabará valiendo también para el reconocimiento del habla. El último 10 por ciento incluye abordar no solo los problemas del ruido, los acentos extraños y las palabras desconocidas, sino también el hecho de que la ambigüedad y la sensibilidad del lenguaje al contexto pueden afectar a su interpretación. ¿Qué hace falta para superar ese último y terco 10 por ciento? ¿Más datos? ¿Más capas de red? ¿O —me atrevo a preguntar— ese último 10 por ciento exigirá comprender realmente lo que está diciendo la persona que habla? Me inclino por esto último, pero no sería la primera vez que me equivoco.

Los sistemas de reconocimiento del habla son bastante complicados; se necesitan varios tipos de procesamiento para pasar de ondas de sonido a frases. Los sistemas de reconocimiento del habla más avanzados en la actualidad integran distintos componentes; entre otros, varias redes neuronales profundas.[230] Otras tareas de PLN, como la traducción de idiomas o la respuesta a preguntas, parecen a primera vista más sencillas: tanto los datos de entrada como los de salida son palabras. Sin embargo, el método del aprendizaje profundo a partir de los datos no ha permitido avanzar tanto en estas áreas como en el reconocimiento del habla. ¿Por qué? Para responder, veamos algunos ejemplos de cómo se ha aplicado el aprendizaje profundo a tareas importantes de PLN.

La clasificación de sentimientos

Como primer ejemplo, veamos el área denominada clasificación de sentimientos. Consideremos estas breves reseñas de la película *Indiana Jones y el templo maldito*:[231]

> «La trama es plúmbea y el sentido del humor se ha perdido».
>
> «Un poco demasiado oscura para mi gusto».
>
> «Parece como si los productores hubieran intentado hacerla lo más perturbadora y terrorífica posible».
>
> «El desarrollo de los personajes y el humor de *El templo maldito* son muy mediocres».
>
> «El tono es un poco raro y tiene mucho humor que a mí no me ha hecho gracia».
>
> «No tiene nada del encanto y el ingenio de las otras de esta serie».

¿Le gustó la película al autor de cada reseña?

Hay mucho dinero invertido en utilizar ordenadores para responder a esa pregunta. Un sistema de IA capaz de clasificar con precisión una frase (o un pasaje más largo) con arreglo a los sentimientos que expresa —positivos, negativos u otro tipo de opinión— sería oro macizo para las empresas

deseosas de analizar los comentarios de los clientes sobre sus productos, encontrar nuevos clientes potenciales, automatizar recomendaciones de productos («a la gente a la que le gustó X también le gusta Y») o dirigir sus anuncios por internet a públicos seleccionados. Los datos sobre las películas, los libros y otros productos que gustan o no a una persona pueden ser sorprendentemente (y tal vez alarmantemente) útiles para predecir sus compras futuras. Es más, esa información puede ayudar a hacer predicciones sobre otros aspectos de la vida de una persona, como sus probables patrones de voto y su receptividad a determinados tipos de noticias o anuncios políticos.[232] Además, ha habido varios intentos, con éxito variable, de utilizar la «extracción de sentimientos» de, por ejemplo, tuits relacionados con la economía, para predecir los precios de las acciones y los resultados electorales.

Dejando al margen el aspecto ético de estas aplicaciones para analizar sentimientos, vamos a centrarnos en cómo podrían los sistemas de IA ser capaces de clasificar los sentimientos expresados en frases como las anteriores. Para un humano es bastante fácil ver que estas minirreseñas son todas negativas, pero conseguir que un programa haga este tipo de clasificación de forma general es mucho más difícil de lo que podría parecer a primera vista.

Algunos de los primeros sistemas de PLN buscaban la presencia de palabras sueltas o secuencias cortas de palabras para detectar los sentimientos de un texto. Por ejemplo, en las críticas de cine, palabras como *oscuro*, *raro*, *plúmbeo*, *perturbador*, *horrible*, *carente* y *perder*, o secuencias como «no me hizo gracia», «no tiene nada», «un poco demasiado», indican un sentimiento negativo. En algunos casos es así, pero en muchos otros este tipo de secuencias también se pueden encontrar en críticas positivas. He aquí algunos ejemplos:

> «A pesar de lo plúmbeo del tema, hay humor suficiente para evitar que resulte demasiado oscuro».

«Aquí no hay nada perturbador ni espantoso como algunos han dado a entender».

«Yo era demasiado joven para ver esta magnífica película cuando se estrenó».

«Si no la ves, tú te lo pierdes».

En general, examinar palabras sueltas o secuencias cortas de forma aislada no basta para captar el sentimiento general; es necesario captar la semántica de las palabras en el contexto de toda la frase.

Poco después de que las redes profundas empezaran a conseguir éxitos en la visión por ordenador y el reconocimiento del habla, los profesionales del PLN probaron a utilizarlas en el análisis de sentimientos. Como de costumbre, se trata de entrenar la red con muchos ejemplos etiquetados por humanos de frases que contienen sentimientos positivos y negativos, y hacer que la propia red aprenda características útiles que le permitan proponer con seguridad una clasificación de «positivo» o «negativo» en una nueva frase. Pero, antes de nada, ¿cómo podemos hacer que una red neuronal procese una frase?

Redes neuronales recurrentes

Para procesar una frase o un fragmento hace falta un tipo de red neuronal diferente de las que he descrito en capítulos anteriores. Recordemos, por ejemplo, la red neuronal convolucional del capítulo 4 que clasificaba una imagen como «perro» o «gato». En ese caso, los datos de entrada de la red eran las distintas intensidades de los píxeles en una imagen de tamaño fijo (las imágenes más grandes o más pequeñas tenían que modificarse para ajustarlas a la escala adecuada). En cambio, las frases están formadas por secuencias de palabras y no tienen una longitud fija. Por eso necesitamos que una red neuronal pueda procesar frases de longitud variable.

La aplicación de redes neuronales a tareas con secuencias ordenadas, como las frases, se remonta a los años ochenta, cuando se introdujeron las redes neuronales recurrentes (RNN, por sus siglas en inglés), inspiradas,

claro está, en las ideas sobre cómo interpreta secuencias el cerebro. Imaginemos que nos piden que leamos la opinión de que es «un poco demasiado oscura para mi gusto» y clasifiquemos el sentimiento que expresa como positivo o negativo. Leemos la frase de izquierda a derecha, palabra por palabra. A medida que la leemos, empezamos a tener una impresión de esos sentimientos, que se confirman al terminar. A esas alturas, nuestro cerebro tiene una especie de representación de la frase en forma de activaciones neuronales, que nos permiten afirmar con seguridad si la crítica es positiva o negativa.

Las redes neuronales recurrentes se inspiran vagamente en este proceso secuencial de leer una frase y crear su representación en forma de activaciones neuronales. La figura 32 compara las estructuras de una red neuronal tradicional y una red neuronal recurrente. Para simplificar, cada red tiene dos unidades (círculos blancos) en la capa oculta y una unidad en la capa de salida. En las dos redes, la entrada tiene conexiones con las unidades ocultas, y cada unidad oculta tiene una conexión con la unidad de salida (flechas continuas). La diferencia fundamental de la RNN es que sus unidades ocultas tienen además unas conexiones «recurrentes»; cada unidad oculta tiene una conexión consigo misma y con la otra unidad oculta (flechas discontinuas). ¿Cómo funciona? A diferencia de una red neuronal tradicional, una RNN actúa en una serie de pasos temporales. En cada paso, la RNN recibe una entrada y calcula la activación de sus unidades ocultas y de salida, igual que hace una red neuronal tradicional. Pero en una RNN cada unidad oculta calcula su activación basándose tanto en la entrada como en las activaciones de las unidades ocultas del paso temporal anterior. (En el primer paso temporal, estos valores recurrentes se fijan en cero). Esto otorga a la red una forma de interpretar las palabras que «lee» mientras recuerda el contexto de lo que ya ha «leído».

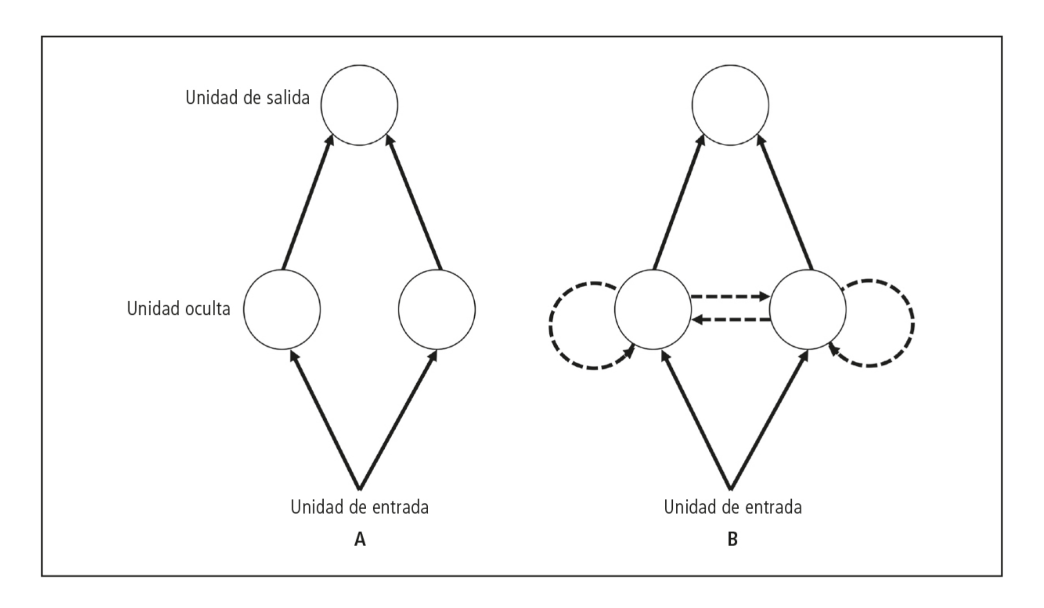


Figura 32. A, ilustración de una red neuronal tradicional; B, ilustración de una red neuronal recurrente, en la que las activaciones de las unidades ocultas en un paso temporal dado se transmiten al siguiente paso temporal.

La mejor forma de entender cómo funcionan las RNN es imaginar la actividad de la red a lo largo del tiempo, como en la figura 33 (en la página siguiente), que muestra la RNN de la figura 32 durante ocho pasos temporales. Para simplificar la ilustración, represento todas las conexiones recurrentes en la capa oculta como una única flecha discontinua de un paso temporal al siguiente. En cada paso temporal, la activación de las unidades ocultas es la codificación por parte de la red de la parte de la frase que ha visto hasta ese momento. La red va refinando esa codificación a medida que procesa las palabras. Después de la última palabra de la frase, recibe un símbolo especial de FIN (equivalente a un punto), que le indica que la frase ha terminado. Hay que tener en cuenta que los humanos añaden el símbolo FIN a cada frase antes de introducir el texto en la red.

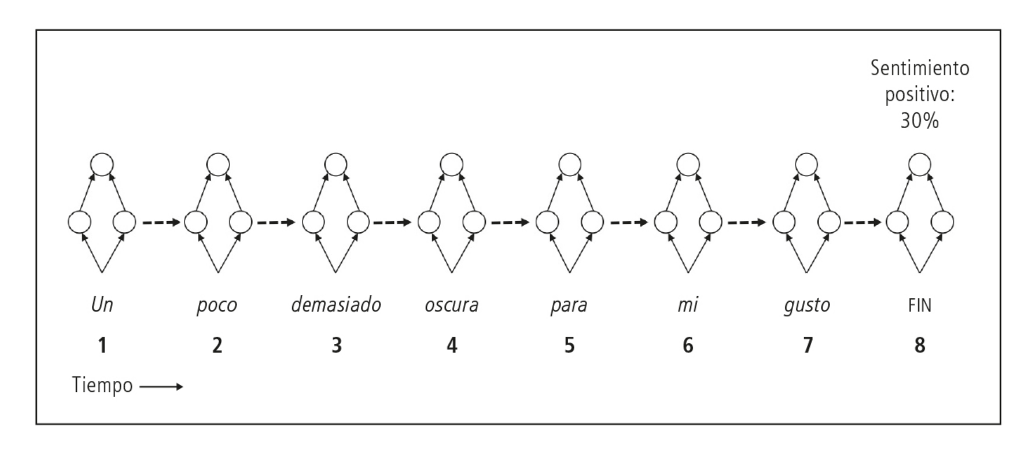


Figura 33. La red neuronal recurrente de la figura 32, funcionando durante ocho pasos temporales.

En cada paso temporal, la unidad de salida de esta red procesa las activaciones de las unidades ocultas (la «codificación») para dar a la red la seguridad de que la frase de entrada (es decir, la parte de la frase introducida en la red hasta ese paso temporal) tiene un sentimiento positivo. Al aplicar la red a una frase determinada, podemos ignorar esta información de salida hasta llegar al final de la frase. Entonces las unidades ocultas codifican toda la frase y la unidad de salida emite la seguridad definitiva de la red (en este caso, el 30 por ciento para el sentimiento positivo o, lo que es lo mismo, el 70 por ciento para el sentimiento negativo).

Como la red solo deja de codificar la frase cuando se encuentra con el símbolo FIN, el sistema, en principio, puede codificar frases de cualquier longitud y convertirlas en un conjunto de números de longitud fija: las activaciones de las unidades ocultas. Por motivos evidentes, este tipo de red neuronal suele denominarse red codificadora.

Con un conjunto de frases que los humanos han etiquetado como sentimientos «positivos» o «negativos», la red codificadora puede entrenarse a partir de estos ejemplos mediante retropropagación. Pero hay algo que aún no he explicado. Las redes neuronales requieren que sus entradas sean números.[233] ¿Cuál es la mejor manera de codificar las palabras de entrada como números? La respuesta a esta pregunta ha

permitido uno de los avances más importantes de la última década en el procesamiento del lenguaje natural.

Un método sencillo para codificar palabras como números

Antes de explicar posibles métodos para codificar palabras en números, tengo que definir el concepto de vocabulario de una red neuronal. El vocabulario es el conjunto de todas las palabras que la red podrá aceptar como entradas. Los lingüistas calculan que un lector necesita entre diez mil y treinta mil palabras para abordar la mayoría de los textos en inglés, dependiendo de cómo se cuenten; por ejemplo, quizá se pueden agrupar las distintas formas de conjugación de un verbo como una sola palabra. El vocabulario también puede incluir términos habituales formados por dos palabras, como *San Francisco* o *Golden Gate*, y contarlos como una sola.

Como ejemplo concreto, supongamos que nuestra red va a tener un vocabulario de veinte mil palabras. El método más sencillo para codificar las palabras como números consiste en asignar a cada palabra del vocabulario un número arbitrario comprendido entre 1 y 20.000. Después damos a la red neuronal 20.000 entradas, una por cada palabra del vocabulario. En cada paso temporal, solo se «activará» una de esas entradas, la correspondiente a la palabra de entrada real. Por ejemplo, supongamos que a la palabra *oscuro* se le ha asignado el número 317. Entonces, si queremos introducir *oscuro* en la red, haremos que la entrada 317 tenga el valor 1 y que las demás 19.999 entradas tengan el valor 0. En el campo del PLN, esto se denomina codificación en caliente: en cada paso temporal, solo una de las entradas —la que corresponde a la palabra que se introduce en la red— está «caliente» (no es un cero).

La codificación en caliente era una forma habitual de introducir palabras en las redes neuronales. Pero tiene un inconveniente: una asignación arbitraria de números a las palabras no captura ninguna relación entre ellas.

Supongamos que la red ha aprendido de sus datos de entrenamiento que la frase «Detesto esta película» expresa un sentimiento negativo. Supongamos ahora que la red recibe como entrada la frase «Aborrezco esta peli», pero no se ha encontrado con «aborrecer» ni con «peli» en sus datos de entrenamiento. La red no tiene forma de determinar que las dos frases quieren decir lo mismo. Supongamos además que la red ha aprendido que la frase «Me reí a carcajadas» está asociada a críticas positivas y que entonces se encuentra con una frase nueva: «Me gustó su humor». La red no sería capaz de reconocer que las dos frases tienen significados cercanos (aunque no exactamente idénticos). La incapacidad de captar las relaciones semánticas entre palabras y frases es una de las principales razones por las que las redes neuronales que utilizan la codificación en caliente no suelen funcionar demasiado bien.

El espacio semántico de las palabras

Los investigadores en PLN han propuesto varios métodos para codificar las palabras de forma que capten esas relaciones semánticas. Todos los métodos se basan en la misma idea, maravillosamente expresada por el lingüista John Firth en 1957: «Dime con quién andas y te diré qué palabra eres».[234] Es decir, el significado de una palabra puede definirse en función de otras palabras con las que suele aparecer, de las palabras que suelen aparecer con esas palabras, y así sucesivamente. *Aborrecer* suele aparecer en los mismos contextos que *detestar*. *Reírse* suele juntarse con las mismas palabras que acompañan a *humor*.

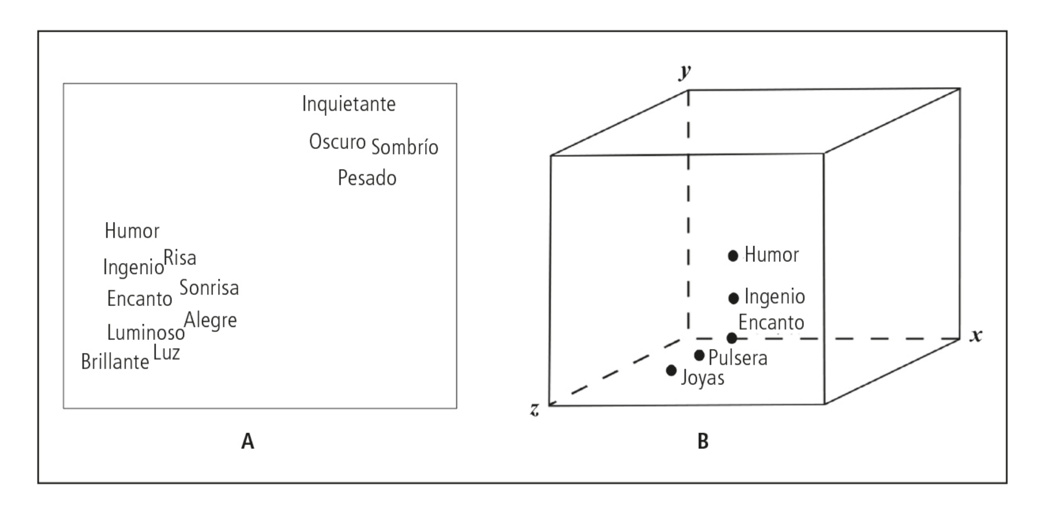


Figura 34. A, ilustración de dos grupos de palabras en un espacio semántico en el que palabras con significados similares se encuentran cerca unas de otras; B, un espacio semántico tridimensional en el que las palabras se representan como puntos.

En lingüística, esta idea recibe el nombre formal de semántica distribucional. La hipótesis básica de la semántica distribucional es que «el grado de similitud semántica entre dos expresiones lingüísticas A y B es una función de la similitud de los contextos lingüísticos en los que A y B pueden aparecer».[235] Los lingüistas suelen concretarlo en la idea de un «espacio semántico». La figura 34A ilustra un espacio semántico bidimensional de palabras en el que las palabras con significados similares están situadas más cerca unas de otras. Pero enseguida se ve que, como las palabras pueden tener muchas dimensiones de significado, su espacio semántico también debe tener más dimensiones. Por ejemplo, la palabra *charm*, «encanto», está cerca de *ingenio* y *humor*, pero en un contexto diferente, *charm* es «colgante» y entonces está cerca de *pulsera* y *joyas*. Del mismo modo, la palabra *brillante* está al lado de los grupos *luminoso* y *alegre*, pero también tiene un significado alternativo (aunque relacionado) que la acerca a *listo*, *inteligente* y *astuto*. Sería útil tener una tercera dimensión que nos transmitiera la página y que situara estas palabras a la distancia justa unas de otras. En una dimensión, *charm* está cerca de *ingenio*, cuando es «encanto»; en otra, está cerca de *pulsera*. Pero *encanto*

también debería estar cerca de *suerte* (mientras que *pulsera* no). Necesitamos más dimensiones. A los seres humanos nos cuesta imaginarnos un espacio de más de tres dimensiones, pero el espacio semántico de las palabras puede necesitar muchas docenas de dimensiones, si no cientos.

Cuando hablamos de espacios semánticos con múltiples dimensiones, entramos en el ámbito de la geometría. De hecho, los profesionales del PLN suelen enmarcar el «significado» de las palabras en conceptos geométricos. Por ejemplo, la figura 34B muestra un espacio tridimensional, con ejes *x*, *y* y *z*, a lo largo del cual se pueden situar las palabras. Cada palabra se identifica con un punto (círculo negro), definido por tres coordenadas, es decir, las posiciones *x*, *y* y *z* del punto. La distancia semántica entre dos palabras se equipara a la distancia geométrica entre los puntos de este gráfico. Se puede ver que *charm* está cerca de *ingenio* y *humor*, y de *pulsera* y *joyas*, pero en dimensiones diferentes. En PLN, se utiliza el término *vector de palabras* para referirse a las coordenadas de una palabra concreta en ese espacio semántico. En matemáticas, *vector* no es más que una palabra sofisticada para referirse a las coordenadas de un punto.[236] Por ejemplo, supongamos que *pulsera* se encuentra en las coordenadas (2, 0, 3); esta lista de tres números es su vector de palabras en este espacio tridimensional. Obsérvese que el número de dimensiones de un vector es simplemente el número de coordenadas.

La idea es que después de que todas las palabras del vocabulario estén colocadas en el lugar debido del espacio semántico, el significado de una palabra puede representarse por su posición en este espacio, es decir, por las coordenadas que definen su vector de palabras. ¿Y para qué sirve un vector de palabras? Resulta que el uso de vectores de palabras como entradas numéricas para representarlas, en contraposición al método de esquema de codificación en caliente que he esbozado antes, mejora mucho el rendimiento de las redes neuronales en tareas de PLN.

¿Cómo se obtienen todos los vectores correspondientes a las palabras de un vocabulario? ¿Existe un algoritmo que coloque correctamente todas las palabras del vocabulario de nuestra red en un espacio semántico para captar mejor las diversas dimensiones del significado de cada palabra? En PLN se ha trabajado mucho para resolver este problema.

Word2vec

Se han sugerido muchas soluciones para el problema de colocar palabras en un espacio geométrico, algunas ya desde los años ochenta, pero el método más generalizado en la actualidad lo propusieron unos investigadores de Google en 2013.[237] Llamaron a su método «word2vec» (la abreviatura de «de palabra a vector»). El método word2vec utiliza una red neuronal tradicional para aprender automáticamente vectores de palabras para todas las palabras de un vocabulario. Los investigadores de Google utilizaron parte del inmenso almacén de documentos de la empresa para entrenar su red; después de completado el entrenamiento, el grupo guardó y publicó todos los vectores de palabras obtenidos en una página web para que cualquiera pudiera descargarlos y utilizarlos como entrada para sistemas de procesamiento del lenguaje natural.[238]

El método word2vec es la encarnación del «dime con quién andas y te diré qué palabra eres». Para crear los datos de entrenamiento del programa word2vec, el grupo de Google empezó por usar un enorme volumen de documentos del servicio Google News. (En el PLN moderno, no hay nada como un montón de «macrodatos»). Los datos de entrenamiento para el programa word2vec consistían en una colección de pares de palabras en los que cada palabra del par había aparecido cerca de la otra en algún sitio de los documentos de Google News. Para que el proceso fuera más eficaz, se descartaron palabras demasiado frecuentes como *el*, *de* e *y*.

Como ejemplo concreto, supongamos que las palabras de cada par aparecen juntas en una frase. En este caso, la frase «un hombre entró en un

restaurante y pidió una hamburguesa» se transformaría primero en «hombre entró en restaurante pidió hamburguesa». Así se obtendrían los siguientes pares: (hombre, entró), (entró, en), (en, restaurante), (restaurante, pidió), (pidió, hamburguesa), además de a la inversa: por ejemplo, (hamburguesa, pidió). Se trata de entrenar a la red word2vec para que prediga qué palabras es probable que se emparejen con una palabra de entrada dada.

La figura 35 ilustra la red neuronal word2vec.[239] Esta red utiliza la codificación en caliente que he descrito antes. En la figura 35 hay setecientas mil unidades de entrada, lo que se aproxima al tamaño del vocabulario utilizado por los investigadores de Google. Cada entrada corresponde a una palabra del vocabulario. Por ejemplo, la entrada 1 corresponde a la palabra *gato*, la 8.378 a *hamburguesa* y la 700.000 a *cerúleo*. Estos números me los acabo de inventar; el orden real no importa. Asimismo, hay setecientas mil unidades de salida, cada una correspondiente a una palabra del vocabulario, y una capa oculta relativamente pequeña de trescientas unidades. Las grandes flechas grises indican que cada entrada tiene una conexión ponderada con cada unidad oculta, y que cada unidad oculta tiene una conexión ponderada con cada unidad de salida.

Los investigadores de Google entrenaron su red con miles de millones de pares de palabras recogidos en artículos de Google News. Dado un par de palabras como (hamburguesa, pidió), la entrada correspondiente a la primera palabra del par (hamburguesa) se establece en uno; todas las demás entradas se establecen en cero. Durante el entrenamiento, la activación de cada unidad de salida se interpreta como la seguridad de la red en que la palabra correspondiente del vocabulario ha aparecido junto a la palabra de entrada. En este caso, las activaciones de salida correctas asignarían una seguridad alta a la segunda palabra del par (pidió).

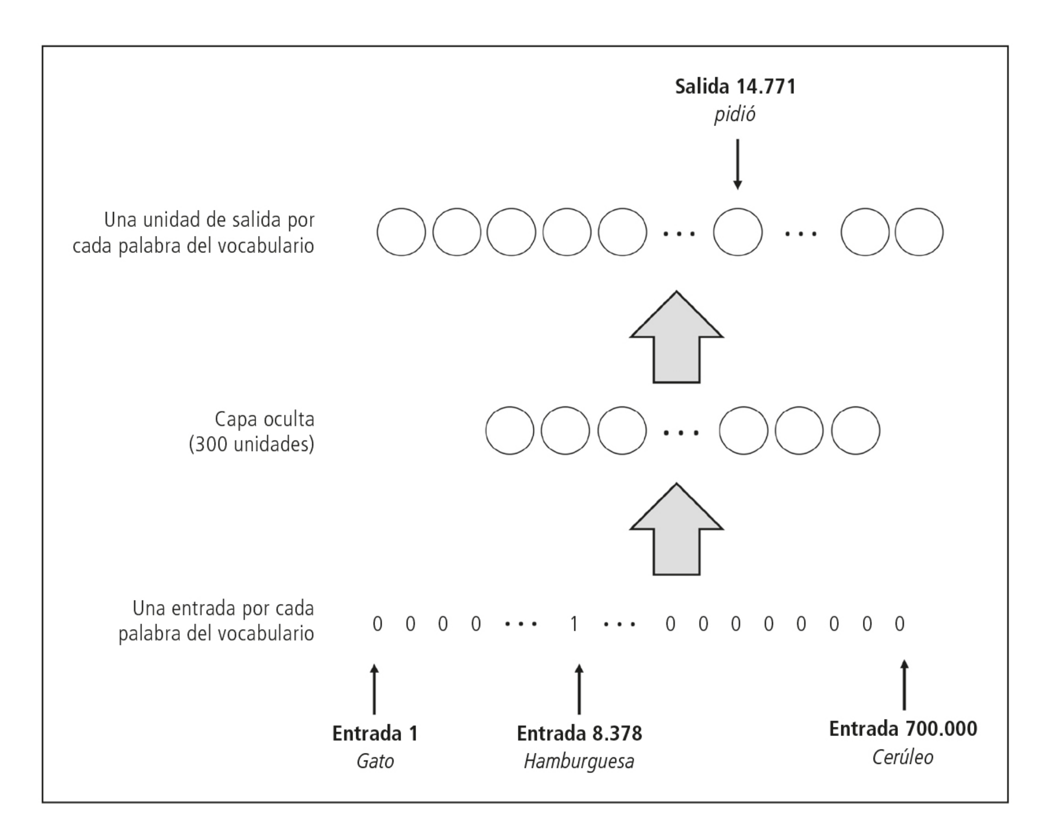


Figura 35. Ilustración de la red neuronal word2vec, dado el par de palabras (hamburguesa, pidió).

Una vez completado el entrenamiento, se puede extrapolar el vector de palabras aprendido a cualquier palabra del vocabulario. La figura 36 (en la página siguiente) ilustra cómo hacerlo. Se ven las conexiones ponderadas entre una entrada (correspondiente a la palabra *hamburguesa*) y las trescientas unidades ocultas. Estos pesos, que se han aprendido a partir de los datos de entrenamiento, han adquirido información sobre los contextos en los que se utiliza la palabra correspondiente. Los trescientos valores de ponderación son los componentes del vector de palabras asignado a la palabra dada. (En este proceso no se tienen en cuenta en absoluto las conexiones entre las unidades ocultas y las salidas; toda la información necesaria está en los pesos de la conexión entre la capa de entrada y la oculta). De modo que los vectores de palabras aprendidos por esta red tienen trescientas dimensiones. La colección de vectores de palabras para

todas las palabras del vocabulario constituye el «espacio semántico» aprendido.

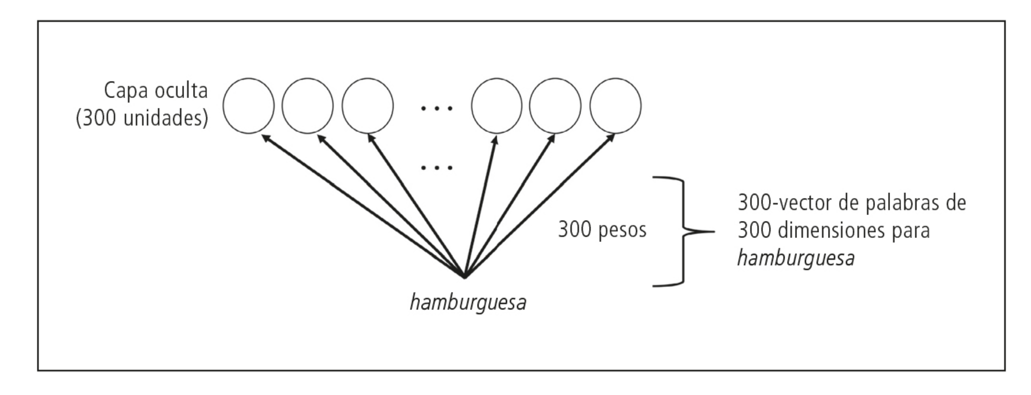


Figura 36. Ilustración de cómo obtener un vector de palabras a partir de la red word2vec entrenada.

Podemos imaginar este espacio semántico tridimensional así. Pensemos en el gráfico tridimensional de la figura 34 e intentemos imaginar un gráfico similar con cien veces más dimensiones y setecientas mil palabras representadas, cada una con trescientas coordenadas. Es broma. Es imposible imaginar algo así.

¿Qué representan estas trescientas dimensiones? Si nosotros fuéramos criaturas tridimensionales con la capacidad mental de imaginar ese espacio, veríamos que cualquier palabra está cerca de otras palabras relacionadas en función de muchos significados. Por ejemplo, el vector de *hamburguesa* está cerca del vector de *pidió*, pero también está cerca de los vectores de *carne*, *perrito caliente*, *vaca*, *comer*, y así sucesivamente. *Hamburguesa* también está cerca de *cena*, aunque nunca haya aparecido en un par con ella; el motivo es que *hamburguesa* está cerca de palabras que también están cerca de *cena* en contextos similares. Si la red ve pares de palabras de las frases «Me comí una hamburguesa para comer» y «Devoré un perrito caliente para cenar», y si *comida* y *cena* también aparecen juntas en otras frases de entrenamiento, el sistema puede aprender que *hamburguesa* y *cena* también deberían estar cerca.

Recordemos que el objetivo de todo este proceso es encontrar una representación numérica —un vector— para cada palabra del vocabulario que capte algo de la semántica de la palabra. La hipótesis es que el uso de estos vectores de palabras producirá redes neuronales de alto rendimiento para tareas de procesamiento del lenguaje natural. ¿Pero hasta qué punto el «espacio semántico» creado por word2vec capta realmente la semántica de las palabras?

Esta pregunta es difícil de responder, porque no podemos imaginar el espacio semántico tridimensional aprendido por word2vec. Pero sí podemos hacer algunas cosas para entrever ese espacio. El método más sencillo consiste en tomar una palabra determinada y encontrar las palabras que han acabado estando más cerca de ella en el espacio semántico a base de observar las distancias entre los vectores de palabras. Por ejemplo, una vez entrenada la red, las palabras más próximas a *Francia* son *España*, *Bélgica*, *Países Bajos*, *Italia*, *Suiza*, *Luxemburgo*, *Portugal*, *Rusia*, *Alemania* y *Cataluña*.[240] Al algoritmo word2vec no se le ha explicado el concepto de *país* o *país europeo*; se trata simplemente de las palabras que aparecen en los datos de entrenamiento en contextos similares a *Francia*, del mismo modo que *hamburguesa* y *perrito caliente* en mi ejemplo anterior. De hecho, si pido las palabras más parecidas a *hamburguesa*, la lista incluye *burger*, *hamburguesa con queso*, *sándwich*, *perrito caliente*, *taco* y *patatas fritas*.[241]

También podemos examinar relaciones más complejas derivadas del entrenamiento de la red. Los investigadores de Google que crearon word2vec vieron que, en los vectores de palabras creados por su red, la distancia entre la palabra para un país y la palabra para la capital de ese país es aproximadamente la misma para muchos países. Se ve en la figura 37 (en la página siguiente), que muestra una representación bidimensional de estas distancias. Tampoco en este caso se introdujo en el sistema el concepto de

capital de un país; estas relaciones surgieron simplemente del entrenamiento de la red con miles de millones de pares de palabras.

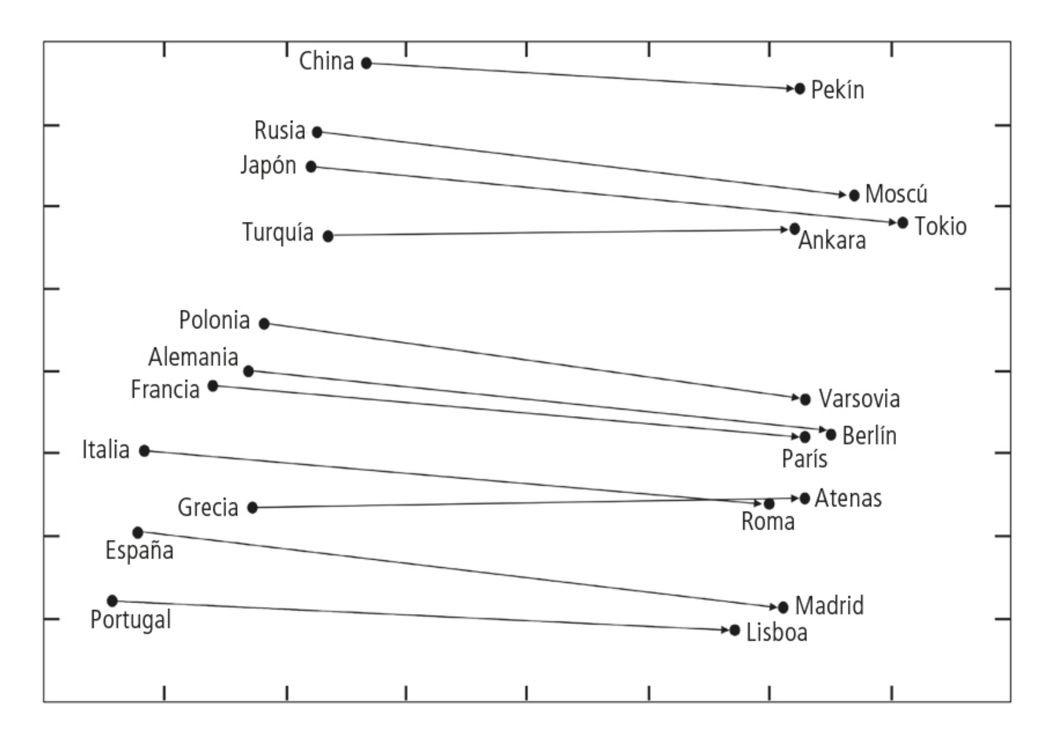


Figura 37. Representación bidimensional de las distancias entre los vectores de palabras de los países y los vectores de palabras de sus capitales.

Este tipo de regularidad hizo pensar a muchos que word2vec podía «resolver» problemas de analogías como «*Hombre* es a *mujer* lo que *rey* es a ______». Basta con tomar el vector de palabras para *mujer*, restar el vector de palabras para *hombre* y sumar el resultado al vector de palabras para *rey*.[242] Sí, es *reina*. En mis experimentos con una demostración de word2vec por internet,[243] este método produce muchas veces resultados muy buenos («*Cena* es a *noche* lo que *desayuno* es a *mañana*»), pero muchas otras veces son crípticos («*Sediento* es a *bebida* lo que *cansado* es a *borracho*») o no tienen sentido («*Pez* es a *agua* lo que *pájaro* es a *boca de riego*»).

Estas propiedades de los vectores de palabras aprendidos son curiosas y demuestran que algunas relaciones se captan. ¿Pero ayudan a que los vectores de palabras sean útiles en general en las tareas de PLN? La

respuesta parece ser un «sí» categórico. Hoy en día, prácticamente todos los sistemas de PLN utilizan vectores de palabras de un tipo u otro (word2vec no es más que una variante) para introducir palabras en el sistema.

Hablando de analogías, aquí hay una: a una persona con un martillo, todo le parece un clavo; a un investigador de IA con una red neuronal, todo le parece un vector. A mucha gente se le ocurrió que el truco de word2vec se podía utilizar no solo con palabras, sino también con frases enteras. ¿Por qué no codificar una frase entera como un vector, del mismo modo que se codifican las palabras, utilizando en el entrenamiento pares de frases en lugar de pares de palabras? ¿No se captaría así la semántica mejor que con un simple conjunto de vectores de palabras? De hecho, varios grupos lo han intentado: un grupo de la Universidad de Toronto llamó a estas representaciones de frases «vectores de pensamiento».[244] Otros han experimentado con redes que codifican como vectores párrafos y documentos enteros, aunque con desigual éxito. Reducir toda la semántica a geometría es una idea atractiva para los investigadores de IA. «Creo que se puede capturar una idea mediante un vector», proclamó Geoffrey Hinton, de Google.[245] Yann LeCun, de Facebook, se mostró de acuerdo: «[En Facebook AI Research] queremos incrustar el mundo en vectores de pensamiento. Lo llamamos World2Vec».[246]

Una última nota sobre los vectores de palabras. Varios grupos han demostrado que estos vectores de palabras, como quizá era previsible, captan los sesgos intrínsecos de los datos lingüísticos que los generan.[247] Por ejemplo, en este problema de analogía: «*Hombre* es a *mujer* lo que *programador informático* es a ______». Si se resuelve utilizando los vectores de palabras que proporciona Google, la respuesta es «ama de casa». El enunciado inverso, «*Mujer* es a *hombre* lo que *programador informático* es a ______», da «ingeniero mecánico». Y aquí hay otro: «*Hombre* es a *genio* lo que *mujer* es a ______». Respuesta: «musa». ¿Y «*Mujer* es a *genio* lo que *hombre* es a ______»? Respuesta: «genios».

Toma ya décadas de feminismo. No podemos culpar a los vectores de palabras; se limitan a captar el sexismo y otros sesgos de nuestro lenguaje, y nuestro lenguaje refleja los prejuicios de nuestra sociedad. Pero, por muy inocentes que sean, los vectores de palabras son un componente fundamental de todos los sistemas modernos de PLN, desde el reconocimiento de voz hasta la traducción de idiomas. Los sesgos en los vectores de palabras pueden deslizarse y generar sesgos inesperados y difíciles de predecir en aplicaciones de PLN muy utilizadas. Los especialistas en IA que investigan estos sesgos están empezando a comprender qué tipo de efectos sutiles pueden tener en los resultados que emiten los sistemas de PLN, y varios grupos están desarrollando algoritmos para «eliminar los sesgos» de los vectores.[248] Eliminar los sesgos de los vectores de palabras es difícil, pero seguramente menos que la alternativa: eliminar los sesgos del lenguaje y la sociedad.

---

[225] Mi historia del «restaurante» está inspirada en pequeñas historias similares creadas por Roger Schank y sus colegas en sus investigaciones sobre la comprensión del lenguaje natural (R. C. Schank y C. K. Riesbeck, *Inside Computer Understanding: Five Programs Plus Miniatures*, Hillsdale, N. J.: Lawrence Erlbaum Associates, 1981) y por John Searle en sus críticas a la IA (J. R. Searle, «Minds, Brains, and Programs», *Behavioral and Brain Sciences* 3, n.º 3 [1980], pp. 417-424).

[226] G. Hinton *et al.*, «Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups», *IEEE Signal Processing Magazine* 29, n.º 6 (2012), pp. 82-97.

[227] J. Dean, «Large Scale Deep Learning», diapositivas de la lección magistral, «Conference on Information and Knowledge Management (CIKM)», noviembre de 2014, consultado el 7 de diciembre de 2018, static.googleusercontent.com/media/research.google.com/en//people/jeff/CIKM-keynote-Nov2014.pdf.

[228] S. Levy, «The iBrain Is Here, and It’s Already in Your Phone», *Wired*, 24 de agosto de 2016, www.wired.com/2016/08/an-exclusive-look-at-how-ai-and-machinelearning-work-at-apple.

[229] En la literatura sobre reconocimiento del habla, la métrica más utilizada para valorar el rendimiento es la «tasa de error de palabras» en grandes colecciones de segmentos cortos de audio. Aunque el rendimiento de los sistemas de reconocimiento del habla más avanzados aplicado a estas colecciones está al «nivel humano» o por encima de él, hay motivos para sostener que cuando se utilizan medidas más realistas (por ejemplo, habla con ruido, con un acento, palabras altisonantes, lenguaje ambiguo), el reconocimiento del habla por parte de las máquinas sigue siendo muy inferior

al de los humanos. Un buen resumen de algunos de estos argumentos se ofrece en A. Hannun, «Speech Recognition Is Not Solved», consultado el 7 de diciembre de 2018, awni.github.io/speech-recognition.

[230] Hay una buena descripción, aunque técnica, del funcionamiento de los algoritmos modernos de reconocimiento del habla en J. H. L. Hansen y T. Hasan, «Speaker Recognition by Machines and Humans: A Tutorial Review», *IEEE Signal Processing Magazine* 32, n.º 6 (2015), pp. 74-99.

[231] Estas reseñas proceden de Amazon.com; en algunos casos, las he editado ligeramente.

[232] En el momento de escribir estas líneas, el mundo de internet sigue conmocionado por la noticia de que una empresa de análisis de datos llamada Cambridge Analytica utilizó datos de decenas de millones de cuentas de Facebook para dirigir propaganda política de forma selectiva, probablemente utilizando métodos de clasificación de sentimientos, entre otras técnicas.

[233] Recordemos (capítulo 2) que cada unidad de una red neuronal calcula una función matemática de la suma de sus entradas multiplicadas por sus pesos. Esto solo puede hacerse si las entradas son números.

[234] J. Firth, «A Synopsis of Linguistic Theory, 1930–1955», en *Studies in Linguistic Analysis*, Oxford: Philological Society, 1957, pp. 1-32.

[235] A. Lenci, «Distributional Semantics in Linguistic and Cognitive Research», *Italian Journal of Linguistics* 20, n.º 1 (2008), pp. 1-31.

[236] En física, el término *vector* suele definirse como una entidad que tiene una magnitud y una dirección. Esta definición es equivalente a la que he dado en el texto: cualquier vector puede describirse particularmente mediante las coordenadas de un punto; la magnitud es la longitud de un segmento desde el origen hasta ese punto, mientras que la dirección es el ángulo que forma ese segmento con los ejes de coordenadas.

[237] T. Mikolov *et al.*, «Efficient Estimation of Word Representations in Vector Space», en *Proceedings of the International Conference on Learning Representations* (2013).

[238] Word2vec, Google Code Archive, code.google.com/archive/p/word2vec/. Los vectores de palabras también se denominan incrustaciones de palabras.

[239] Aquí estoy ilustrando una versión del método *skip-gram* (n-grama con saltos), que fue uno de los dos métodos propuestos en Mikolov *et al.*, «Efficient Estimation of Word Representations in Vector Space».

[240] *Ibid.*

[241] He utilizado la demo word2vec en bionlp-www.utu.fi/wvdemo/ (con el modelo «English Google-News Negative300») para obtener estos resultados.

[242] La idea es resolver x en el problema de aritmética vectorial $hombre - mujer = = rey - x$. Para sumar o restar dos vectores, basta con sumar o restar sus elementos correspondientes; por ejemplo, $(3, 2, 4) - (1, 1, 1) = (2, 1, 3)$.

[243] bionlp-www.utu.fi/wv_demo/.

[244] R. Kiros *et al.*, «Skip-Thought Vectors», en *Advances in Neural Information Processing Systems* 28 (2015), pp. 3294-302.

[245] Citado en H. Devlin, «Google a Step Closer to Developing Machines with Human-Like Intelligence», *The Guardian*, 21 de mayo de 2015, www.theguardian.com/science/2015/may/21/google-a-step-closer-to-developing-machines-with-humanlike-intelligence.

[246] Y. LeCun, «What's Wrong with Deep Learning?», diapositivas de conferencias, p. 77, consultado el 14 de diciembre de 2018, www.pamitc.org/cvpr15/files/lecun-20150610-cvpr-keynote.pdf.

[247] Véase, por ejemplo, T. Bolukbasi *et al.*, «Man Is to Computer Programmer as Woman Is to Homemaker? Debiasing Word Embeddings», en *Advances in Neural Information Processing Systems* 29 (2016), pp. 4349-4357.

[248] Véase, por ejemplo, J. Zhao *et al.*, «Learning Gender-Neutral Word Embeddings», en *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (2018), pp. 4847-4853, y A. Sutton, T. Lansdall-Welfare y N. Cristianini, «Biased Embeddings from Wild Data: Measuring, Understanding, and Removing», en *Proceedings of the International Symposium on Intelligent Data Analysis* (2018), pp. 328-339.

12

La traducción como codificación y descodificación

Si alguna vez han utilizado Google Translate o cualquier otro sistema moderno de traducción automática, sabrán que el sistema puede traducir un texto de un idioma a otro en una fracción de segundo. Lo que es aún más impresionante es que los sistemas de traducción en línea proporcionan estas traducciones en fracciones de segundo a gente de todo el mundo, veinticuatro horas al día, siete días a la semana, y casi todos pueden hacerlo con más de cien idiomas diferentes. Hace varios años, cuando mi familia y yo vivimos en Francia durante seis meses sabáticos, utilicé Google Translate para redactar correos electrónicos escrupulosamente diplomáticos a nuestra casera francesa, que era muy formal, sobre un problema de humedades en la casa. Como mi francés no era ni mucho menos perfecto, Google Translate me ahorró horas de buscar palabras que no conocía, por no hablar de intentar recordar dónde van los acentos y de qué género es cada sustantivo.

También usé Google Translate para interpretar las respuestas a menudo confusas de la casera, y, aunque las traducciones del programa me daban una idea bastante clara de lo que quería decir, el inglés que utilizaba estaba lleno de errores, grandes y pequeños. Todavía me estremezco cuando imagino lo que le parecerían a ella mis mensajes en francés. En 2016,

Google lanzó un nuevo sistema de «traducción automática neuronal» que, según la empresa, ha logrado «las mayores mejoras en la traducción automática hasta la fecha»,[249] pero la calidad de los sistemas de traducción automática sigue siendo muy inferior a la de los buenos traductores humanos.

Espoleada en parte por la Guerra Fría entre Estados Unidos y la Unión Soviética, la traducción automática —sobre todo entre inglés y ruso— fue uno de los primeros proyectos de IA. En 1947, el matemático Warren Weaver promovió con entusiasmo los primeros métodos: «Es natural preguntarse si el problema de la traducción podría abordarse como un problema de criptografía. Cuando veo un artículo en ruso, digo: "Esto está escrito en inglés, pero está codificado con unos símbolos extraños. Voy a descodificarlo"».[250] Como es habitual en la IA, esa «descodificación» acabó siendo más difícil de lo que la gente esperaba en un principio.

Al igual que ocurrió con otras investigaciones sobre IA en los primeros tiempos, los métodos originales de traducción automática se basaban en complicados conjuntos de reglas especificadas por el ser humano. Para traducir de una lengua de partida (por ejemplo, el inglés) a una de llegada (por ejemplo, el ruso), se le proporcionaba al sistema de traducción automática un conjunto de reglas sintácticas de ambas lenguas, así como las reglas de correspondencia entre estructuras sintácticas. Además, los programadores humanos creaban diccionarios para el sistema de traducción automática con equivalencias entre palabras (y frases sencillas). Como pasó con muchas otras iniciativas de IA simbólica, aunque estos métodos funcionaban bien en algunos casos concretos, eran bastante frágiles y sufrían todos los problemas del lenguaje natural que he expuesto antes.

A partir de los años noventa empezó a dominar este terreno un nuevo método, denominado traducción automática estadística. Siguiendo la tendencia de la IA en aquella época, la traducción automática estadística se basaba en el aprendizaje a partir de datos, en lugar de que los humanos

especificaran unas reglas. Los datos de entrenamiento consistían en grandes colecciones de pares de frases: la primera frase de cada par procedía de la lengua de partida, y la segunda era una traducción (creada por un humano) a la lengua de llegada. Estos pares de frases se extraían de documentos oficiales de países bilingües (por ejemplo, todos los documentos del Parlamento canadiense se elaboran en inglés y francés), transcripciones de Naciones Unidas, que se traducen a las seis lenguas oficiales de la ONU, y otros grandes conjuntos de documentos originales y traducidos.

Los sistemas estadísticos de traducción automática de los años noventa y dos mil solían calcular grandes tablas de probabilidades que relacionaban frases en las lenguas de partida y de llegada. Cuando se proporcionaba al sistema una nueva frase en inglés —por ejemplo, «A man went into a restaurant»—, el sistema dividía la frase en «sintagmas» («A man went», «into a restaurant») y buscaba en sus tablas de probabilidades las mejores traducciones en la lengua de llegada. Estos sistemas tenían más pasos para asegurarse de que todos los sintagmas traducidos funcionaban juntos como oración, pero el motor principal de la traducción eran las probabilidades de los sintagmas aprendidos a partir de los datos de entrenamiento. Aunque los sistemas estadísticos de traducción automática apenas conocían la sintaxis de ninguno de los dos idiomas, en general estos métodos producían mejores traducciones que los primeros métodos basados en reglas.

Google Translate —probablemente el programa de traducción automática más utilizado— empleó este tipo de métodos estadísticos de traducción automática desde su lanzamiento, en 2006, hasta 2016, cuando sus investigadores desarrollaron un método superior de traducción basado en el aprendizaje profundo y denominado traducción automática neuronal. Poco después, la traducción automática neuronal se adoptó en todos los programas de traducción automática de última generación.

Codificador, te presento al descodificador

En la figura 38 (en la página siguiente) se muestra un esbozo de lo que ocurre cuando se utiliza Google Translate (y otros programas actuales de traducción automática) para traducir del inglés al francés.[251] Es un sistema complicado y he simplificado muchos detalles, pero esta figura permite ver las ideas fundamentales.[252]

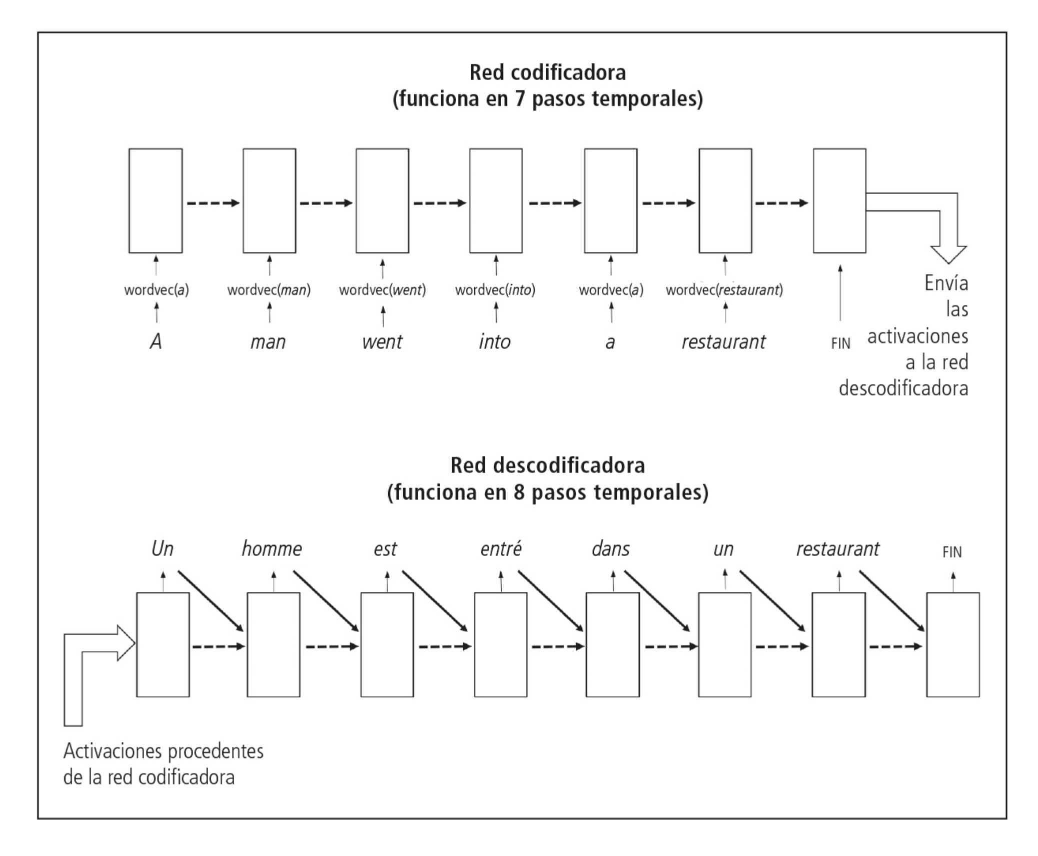


Figura 38. Esquema de un par de redes «codificador-descodificador» para la traducción de idiomas. Los rectángulos blancos representan las redes codificadora y descodificadora, que operan en pasos temporales sucesivos. Las palabras de entrada —por ejemplo, man— se convierten primero en vectores de palabras —por ejemplo, wordvec(man)— antes de pasar a la red.

La mitad superior de la figura 38 muestra una red neuronal recurrente (una red codificadora), muy parecida a la que describí en el capítulo anterior. La frase en inglés «A man went into a restaurant» se codifica en siete pasos temporales. He usado rectángulos blancos para representar la red que codifica esta frase; un poco más adelante hablaré de cómo es la red

dentro de los rectángulos. Durante la etapa de codificación, en cada paso temporal se introduce en la red una palabra de la frase en forma de vector de palabras como los que describí antes.[253] Las flechas discontinuas de un paso temporal al siguiente indican las conexiones recurrentes en la capa oculta. La red construye palabra a palabra una representación de la frase en inglés, codificada en la activación de sus unidades ocultas.

En el último paso temporal, la red codificadora recibe el símbolo FIN y las activaciones de las unidades ocultas se convierten en la codificación de la frase. Estas activaciones definitivas de las unidades ocultas del codificador son los datos de entrada a una segunda red, descodificadora, que crea la versión traducida de la frase. La red descodificadora, ilustrada en la mitad inferior de la figura 38, no es más que otra red recurrente, pero en la que las salidas son números que representan las palabras que forman la frase traducida, cada una de las cuales se vuelve a introducir en la red en el siguiente paso temporal.[254]

Hay que señalar que la frase francesa tiene siete palabras, mientras que la inglesa tiene seis. En teoría, este sistema codificador-descodificador puede traducir una frase de cualquier longitud a otra de cualquier otra longitud.[255] Sin embargo, cuando las frases son demasiado largas, la red codificadora acaba perdiendo información útil; es decir, en los pasos temporales más avanzados «olvida» partes anteriores importantes de la frase. Por ejemplo, veamos esta frase:

> Mi madre dice que el gato que voló con su hermana a Hawái el año antes de que empezaras en ese nuevo instituto está viviendo ahora con mi primo.

¿Quién vive con mi primo? En algunos idiomas, la respuesta puede modificar la traducción de los verbos *está* y *viviendo*. A los humanos se les da bien procesar este tipo de frases enrevesadas, pero las redes neuronales recurrentes pueden perder el hilo con facilidad. Cuando la red intenta codificar toda la frase en una serie de activaciones de unidades ocultas, la cosa se embarulla.

A finales de los años noventa, un grupo de investigación suizo propuso una solución: las unidades individuales de una red neuronal recurrente debían tener una estructura más complicada, con pesos especializados que determinaran qué información se transmite en el siguiente paso temporal y qué información puede «olvidarse». A las unidades más complejas las llamaron unidades de «memoria a corto plazo de larga duración» (LSTM, por sus siglas en inglés).[256] El nombre es confuso, pero la idea es que estas unidades permiten más memoria «a corto plazo» capaz de durar durante todo el procesamiento de la frase. Los pesos especializados se aprenden mediante retropropagación, al igual que los pesos normales de una red neuronal tradicional. Aunque la figura 38 muestra las redes codificadoras y descodificadoras de forma abstracta, como rectángulos blancos, en realidad estas redes están formadas por unidades de LSTM.

La traducción automática en la era del aprendizaje profundo es un triunfo de los macrodatos y la rapidez de los ordenadores. Para crear dos redes codificadora-descodificadora que traduzcan, por ejemplo, del inglés al francés, las redes se entrenan con más de treinta millones de pares de frases traducidas por humanos. Las redes neuronales recurrentes profundas, formadas por unidades de LSTM y entrenadas con grandes conjuntos de datos, se han convertido en algo cotidiano en los sistemas modernos de procesamiento del lenguaje natural, no solo en las redes de codificación y descodificación utilizadas por Google Translate, sino también en el reconocimiento del habla, la clasificación de sentimientos y, como veremos más adelante, la respuesta a preguntas. Estos sistemas suelen incorporar varios trucos para mejorar su comportamiento, como introducir la frase original hacia delante y hacia atrás, y mecanismos para centrar la atención en distintas partes de la frase en diferentes pasos temporales.[257]

Una evaluación de la traducción automática

Cuando Google Translate puso en marcha su traducción automática neuronal en 2016, la empresa afirmó que el nuevo método estaba «salvando la distancia entre la traducción humana y la automática».[258] Otras grandes empresas tecnológicas se apresuraron a ponerse al día y crearon sus propios programas de traducción automática en línea, también basados en la arquitectura de codificador y descodificador que he descrito. Las empresas y los medios tecnológicos que informan sobre ellas se han mostrado muy entusiastas sobre estos servicios de traducción. La revista *Technology Review* del MIT informó de que «el nuevo servicio de Google traduce idiomas casi tan bien como los humanos».[259] Microsoft anunció en un comunicado de prensa que su servicio de traducción de noticias del chino al inglés había alcanzado «la paridad humana».[260] IBM declaró que «IBM Watson ya domina nueve idiomas (y suma y sigue)».[261] El directivo de Facebook responsable de la traducción de idiomas afirmó en un acto público: «Creemos que las redes neuronales están aprendiendo el significado semántico subyacente del idioma».[262] El director general de la empresa de traducción especializada DeepL presumió: «Nuestras redes neuronales [de traducción automática] han desarrollado un asombroso sentido de la comprensión».[263]

En general, estas declaraciones están impulsadas en parte por la rivalidad entre las empresas tecnológicas para vender distintos servicios de IA a otras empresas, y la traducción de idiomas es una propuesta importante que puede aportar grandes beneficios. Si bien sitios web como Google Translate ofrecen traducción gratuita para textos breves, una empresa que quiera traducir un gran volumen de documentos o suministrar traducción a los clientes en su web puede encontrar muchos servicios de traducción automática de pago, todos basados en la misma arquitectura de codificador-descodificador.

¿Hasta qué punto debemos creer las afirmaciones de que las máquinas están verdaderamente aprendiendo el «significado semántico» o de que la

traducción automática se aproxima rápidamente a los niveles de precisión humana? Para responder a esta pregunta, vamos a analizar con detalle los resultados reales en los que se basan estas afirmaciones. En concreto, cómo miden estas empresas la calidad de una traducción automática o humana. Medir la calidad de una traducción no es nada fácil: hay muchas maneras de traducir bien un texto (y muchas más de traducirlo mal). Como no existe una única respuesta correcta para la traducción de un texto, es difícil diseñar un método automático para calcular la precisión del sistema.

Las expresiones como «paridad humana» y «salvando la distancia entre las máquinas y los seres humanos» en la traducción automática parten de dos métodos de evaluación de las traducciones. El primero es un método automático —un programa informático— que compara la traducción de un ordenador con las de los humanos y emite una puntuación. El segundo método emplea a personas bilingües para que evalúen manualmente las traducciones. En el primer método, el programa utilizado en casi todas las evaluaciones de traducción automática se llama *bilingual evaluation understudy* (sustituto de evaluación bilingüe) o BLEU.[264] Para medir la calidad de una traducción, BLEU cuenta el número de coincidencias entre palabras y frases de longitud variable en una frase traducida por la máquina y una o más traducciones «de referencia» (es decir, «correctas») hechas por humanos. Aunque las puntuaciones obtenidas por BLEU suelen estar en proporción con la valoración humana de la calidad de la traducción, BLEU tiende a sobrevalorar las malas traducciones. Varios investigadores sobre traducción automática me han dicho que BLEU no es un buen método para evaluar traducciones, y que se utiliza solo porque nadie ha encontrado todavía un método automático que funcione mejor en general.

Dados los inconvenientes de BLEU, la referencia para evaluar un sistema de traducción automática es que personas bilingües califiquen de forma manual las traducciones generadas por el sistema. Esos mismos evaluadores humanos también pueden puntuar las traducciones correspondientes creadas

por traductores humanos profesionales para compararlas con las puntuaciones de la traducción automática. Pero este método de referencia también tiene sus inconvenientes: contratar a personas cuesta dinero y, a diferencia de los ordenadores, los humanos se cansan cuando llevan más de unas decenas de frases calificadas. Por tanto, a menos que se pueda contratar a un ejército de evaluadores humanos bilingües con mucho tiempo por delante, el proceso de evaluación será limitado.

Los grupos de traducción automática de Google y Microsoft llevaron a cabo este tipo de evaluación de referencia (aunque limitada) contratando a pequeños grupos de evaluadores humanos bilingües para que dieran sus calificaciones.[265] A cada evaluador se le dio un conjunto de frases en una lengua de partida y las traducciones de esas frases en la lengua de llegada. Las traducciones las habían hecho tanto el sistema neuronal de traducción automática como traductores humanos profesionales. La evaluación de Google consistió en aproximadamente quinientas frases sacadas de noticias y artículos de Wikipedia en varios idiomas. Al promediar las puntuaciones de cada evaluador para todas las frases y después entre todos los evaluadores, los investigadores de Google descubrieron que la puntuación media otorgada a su sistema neuronal de traducción automática se aproximaba (aunque se quedaba por debajo) a las de las frases traducidas por humanos. Ocurrió lo mismo con todos los pares de idiomas de la evaluación.

Microsoft utilizó un método similar de promedios para evaluar las traducciones de noticias del chino al inglés. Las calificaciones de las traducciones hechas por el sistema de traducción automática neuronal de la empresa se acercaron mucho (y a veces incluso superaron) a las de las traducciones humanas. En todos los casos, los evaluadores humanos dieron a las traducciones hechas mediante traducción automática neuronal mejores puntuaciones que a las de los métodos de traducción automática anteriores.

En resumen, con la introducción del aprendizaje profundo, la traducción automática ha mejorado. ¿Pero podemos interpretar que estos resultados justifican la afirmación de que la traducción automática se acerca ya al «nivel humano»? Creo que esta frase no está justificada por varios motivos. En primer lugar, hacer una media de las puntuaciones puede ser engañoso. Imaginemos un caso en el que aunque la mayoría de las traducciones de frases se califican como «estupendas», hay muchas que se califican como «horribles». El promedio sería «bastante buenas». Sin embargo, seguramente preferiríamos un sistema de traducción más fiable, que fuera siempre «bastante bueno» y nunca «horrible».

Además, las afirmaciones de que estos sistemas de traducción se acercan al «nivel humano» o a la «paridad humana» se basan exclusivamente en la evaluación de traducciones de frases sueltas y aisladas, no de pasajes más largos. Las frases de un pasaje más largo pueden depender unas de otras en aspectos importantes que quizá pasan inadvertidos si se traducen de forma aislada. No he visto ningún estudio formal sobre la evaluación de la traducción automática de pasajes más largos, pero mi experiencia general es que la calidad de la traducción de, por ejemplo, Google Translate disminuye de forma considerable cuando se le dan párrafos enteros en lugar de frases sueltas.

Por último, todas las frases de estas evaluaciones proceden de noticias y páginas de Wikipedia, que suelen estar redactadas con cuidado de evitar el lenguaje ambiguo y los modismos; un tipo de lenguaje que puede causar graves problemas a los sistemas de traducción automática.

Lo que se pierde en la traducción

¿Recuerdan la historia del restaurante que conté al principio del capítulo anterior? No la pensé para probar sistemas de traducción, pero es un buen ejemplo para ilustrar los problemas que plantea el lenguaje coloquial, idiomático y quizá ambiguo a los sistemas de traducción automática.

He usado Google Translate para traducir la historia del restaurante del inglés a tres idiomas de llegada: francés, italiano y chino. Entregué las traducciones resultantes (sin el texto original) a varios amigos que son bilingües en inglés y la otra lengua y les pedí que tradujeran la traducción que había hecho Google al inglés, para hacerme una idea de lo que un hablante de la lengua de llegada sacaría en limpio del texto traducido a esa lengua. Aquí están los resultados, para que los disfruten. (Las traducciones de Google Translate que mis amigos volvieron a traducir al inglés figuran en las notas al final del libro). La anécdota original (traducida al español) era:

> Un hombre entró en un restaurante y pidió una hamburguesa poco hecha. Cuando esta llegó a la mesa, estaba completamente quemada. La camarera se acercó. «¿Está bien la carne?», le preguntó. «Está estupenda», dijo el hombre, mientras empujaba la silla hacia atrás y se iba del restaurante hecho una furia y sin pagar. La camarera le gritó: «Eh, ¿y la cuenta?». Después se encogió de hombros y murmuró: «¿Por qué se ha puesto así?».

Versión francesa de Google Translate, traducida por humanos al inglés (en su equivalente español):

> Un hombre entró en un restaurante y pidió una hamburguesa, cocinada con poca frecuencia. Cuando llegó, se quemó en un crocante. La camarera se detuvo frente a la mesa del hombre. «¿Se encuentra bien la hamburguesa?», preguntó. «Está buenísima», dijo el hombre mientras ponía la silla atrás y salía del restaurante sin pagar. La camarera le gritó: «Dígame, ¿qué le parece el proyecto de ley?». Después se encogió de hombros, mientras murmuraba: «¿Por qué está tan deforme?».[266], [267]

Versión italiana de Google Translate, traducida por humanos al inglés (en su equivalente español):

> Un hombre fue a un restaurante y pidió una hamburguesa hecha escasa. Cuando llegó, estaba quemada por un guirlache. La camarera se detuvo cerca de la mesa del hombre. «¿Está bien la hamburguesa?», le preguntó. «Es sencillamente fantástica», dijo el hombre, mientras echaba hacia atrás la silla y salía del restaurante sin pagar. La camarera le gritó: «Eh, ¿y la cuenta?». Después se encogió de hombros y murmuró en voz baja: «¿Por qué está tan doblado?».[268]

Versión china de Google Translate, traducida por humanos al inglés (en su equivalente español):

> Un hombre entró en un restaurante y pidió una hamburguesa poco habitual. Cuando llegó a su destino, estaba tostada, muy crujiente. La camarera se detuvo junto a la mesa del hombre. «¿Está buena la hamburguesa?», le preguntó. «Está buenísima», dijo el hombre, mientras apartaba la silla y salía corriendo del restaurante sin pagar. La camarera gritó: «Eh, ¿y la cuenta?». Después se encogió de hombros y susurró: «¿Por qué estaba tan encorvado?».[269]

Leer estas traducciones es como escuchar una obra musical que conocemos bien interpretada por un pianista de talento pero propenso a cometer errores. La pieza se reconoce en general, pero tiene algunos destrozos incómodos; la melodía avanza de forma maravillosa durante breves instantes, pero hay notas discordantes y equivocadas que no dejan de interrumpirla.

Como ven, Google Translate elige a veces el significado que no es de palabras ambiguas, como *rare* y *bill* (traducidas al francés como «poco frecuente» en vez de «poco hecha», y «proyecto de ley» en vez de «cuenta», respectivamente); el motivo es que el programa ignora el contexto de palabras u oraciones anteriores. Frases hechas como «burn to a crips» (quemada) y «bent out of shape» (molesto) se traducen de forma extraña; parece que el programa no tiene forma de encontrar la expresión correspondiente en la lengua de llegada ni de captar su verdadero significado. Aunque el sentido de la anécdota se conserva, en todas las traducciones se pierden matices sutiles pero importantes, como el enfado del hombre, expresado en «salir hecho una furia del restaurante», y el descontento de la camarera, expresado en «murmurar». Y, por si fuera poco, la gramática correcta brilla a veces por su ausencia.

No quiero criticar especialmente a Google Translate; he probado otros servicios de traducción en línea y los resultados son similares. No es extraño, porque todos estos sistemas utilizan prácticamente la misma arquitectura de codificador-descodificador. También es importante señalar que las traducciones que obtuve no son más que una instantánea de estos sistemas de traducción en el tiempo; están mejorándolos sin cesar y algunos

de los errores de traducción concretos que aparecen aquí pueden estar corregidos para cuando estén leyendo este libro. No obstante, no creo que la traducción automática llegue a alcanzar el nivel de los traductores humanos —salvo en contadas circunstancias— hasta dentro de mucho tiempo.

El principal obstáculo es el siguiente: al igual que los sistemas de reconocimiento del habla, los sistemas de traducción automática llevan a cabo su tarea sin comprender verdaderamente el texto que están procesando.[270] En la traducción, como en el reconocimiento del habla, la pregunta sigue siendo la misma: ¿hasta qué punto es necesaria esa «comprensión» para que las máquinas alcancen niveles humanos de rendimiento? Douglas Hofstadter sostiene: «La traducción es mucho más compleja que un mero buscar en el diccionario y reordenar las palabras. […] Traducir implica tener un modelo mental del mundo del que se habla».[271] Por ejemplo, un ser humano que tradujera la historia del restaurante tendría un modelo mental en el que cuando un hombre se marcha de un restaurante sin pagar, hay más probabilidades de que una camarera le grite por la cuenta de su comida que por el «proyecto de ley». Las palabras de Hofstadter encuentran eco en un artículo reciente de los investigadores de IA Ernest Davis y Gary Marcus: «La traducción automática […] plantea muchas veces problemas de ambigüedad que solo pueden resolverse con una verdadera comprensión del texto y con el conocimiento del mundo real».[272]

¿Podría una red de codificador-descodificador conseguir los modelos mentales y el conocimiento del mundo real necesarios solo con tener un conjunto de datos de entrenamiento mayor y más capas de red, o hace falta algo fundamentalmente distinto? Esta sigue siendo una cuestión sin resolver que suscita intensos debates en el mundo de la IA. De momento, me limitaré a decir que aunque la traducción automática neuronal puede ser tremendamente eficaz y útil en muchas aplicaciones, los resultados, si no los poseditan expertos humanos, siguen siendo poco fiables. Quien recurra

a la traducción automática —y yo misma lo hago— debe tomarse los resultados con reservas. De hecho, cuando le he pedido a Google Translate que tradujera «take it with a grain of salt» (tómalo con reservas) del inglés al chino y luego de nuevo al inglés, me ha dicho que «trajera una barra de sal». Quizá sea mejor idea.

La traducción de imágenes a frases

He aquí una idea absurda: además de traducir entre idiomas, ¿sería posible entrenar algo como un par de redes neuronales codificadoras-descodificadoras para traducir de imágenes a texto? Se trataría de utilizar una red para codificar una imagen y otra red para «traducir» esa imagen a una frase que describa el contenido de la imagen. Al fin y al cabo, ¿crear un pie de foto no es otro tipo de «traducción», entre el «idioma» de una imagen y el idioma de un pie de foto?

Resulta que esta idea no es tan absurda. En 2015, dos grupos —uno de Google y otro de la Universidad de Stanford— publicaron de forma independiente artículos muy similares sobre este tema, en la misma conferencia sobre visión por ordenador.[273] Voy a describir aquí el sistema desarrollado por el grupo de Google, llamado Show and Tell (Muestra y Explica), porque conceptualmente es un poco más sencillo.

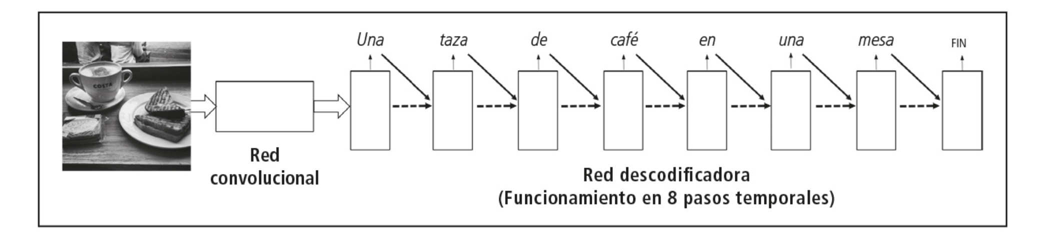


Figura 39. Esquema del sistema automatizado de subtitulación de imágenes de Google.

La figura 39 muestra de forma esquemática el funcionamiento del sistema Show and Tell.[274] Es parecido al sistema codificador-descodificador de la figura 38 (en la página 252), pero aquí la entrada es

una imagen en lugar de una frase. La imagen se introduce en una red neuronal convolucional profunda en lugar de en una red codificadora. Esta ConvNet es similar a las que describí en el capítulo 4, salvo que no emite clasificaciones de objetos, sino que las activaciones de su última capa son los datos de entrada de la red descodificadora. La red descodificadora «descodifica» esas activaciones para dar como resultado una frase. Para codificar la imagen, los autores usaron una ConvNet a la que se había entrenado para clasificar imágenes en ImageNet, el enorme conjunto de datos de imágenes que describí en el capítulo 5. En este caso, la tarea consiste en entrenar a la red descodificadora para que genere un pie de foto apropiado para la imagen de entrada.

¿Cómo aprende este sistema a producir pies que tengan sentido? Recordemos que en el caso de la traducción de idiomas, los datos de entrenamiento consisten en pares de frases en los que la primera frase está en la lengua de partida y la segunda es la traducción hecha por un humano a la lengua de llegada. En el caso de los pies de imágenes, cada ejemplo de entrenamiento consiste en una imagen emparejada con un pie de foto. Las imágenes se descargaron de depósitos como Flickr.com, y la elaboración de los pies de foto de esas imágenes corrió a cargo de seres humanos, en concreto empleados de Amazon Mechanical Turk, contratados por Google para este estudio. Como los pies de foto pueden ser muy variables, para cada imagen hicieron los correspondientes pies cinco personas distintas. Por tanto, cada imagen aparece cinco veces en el conjunto de entrenamiento, cada vez con un pie de foto distinto. En la figura 40 se ve una imagen de entrenamiento de muestra y los pies de foto proporcionados por los empleados de Mechanical Turk.



Una taza caliente y una tostada para comer.

Un sándwich a la plancha y una bebida de café en una mesa.

Un plato blanco con tostadas y una taza de café.

Una taza de café junto a un panini sobre una encimera.

En la mesa hay un bocadillo, una taza de café y una galleta.

Figura 40. Ejemplo de imagen de entrenamiento con pies de foto proporcionados por empleados de Amazon Mechanical Turk.

La red de descodificación Show and Tell se entrenó aproximadamente con ochenta mil pares de imágenes y pies explicativos. En la figura 41 se ven algunos ejemplos de pies que el sistema, una vez entrenado, generó en imágenes de prueba, es decir, imágenes que no estaban en su conjunto de entrenamiento.

Es difícil no sentirse deslumbrados y quizá un poco asombrados por el hecho de que una máquina pueda recoger imágenes en forma de píxeles en bruto y generar unos pies de foto tan precisos. Así me sentí cuando leí por primera vez sobre estos resultados en *The New York Times*. El autor del artículo, el periodista John Markoff, hizo una cautelosa descripción: «Dos grupos de científicos, trabajando de forma independiente, han creado un programa de inteligencia artificial capaz de identificar y describir el contenido de fotografías y vídeos con mucha más precisión que nunca, a veces incluso emulando los niveles humanos de comprensión».[275]

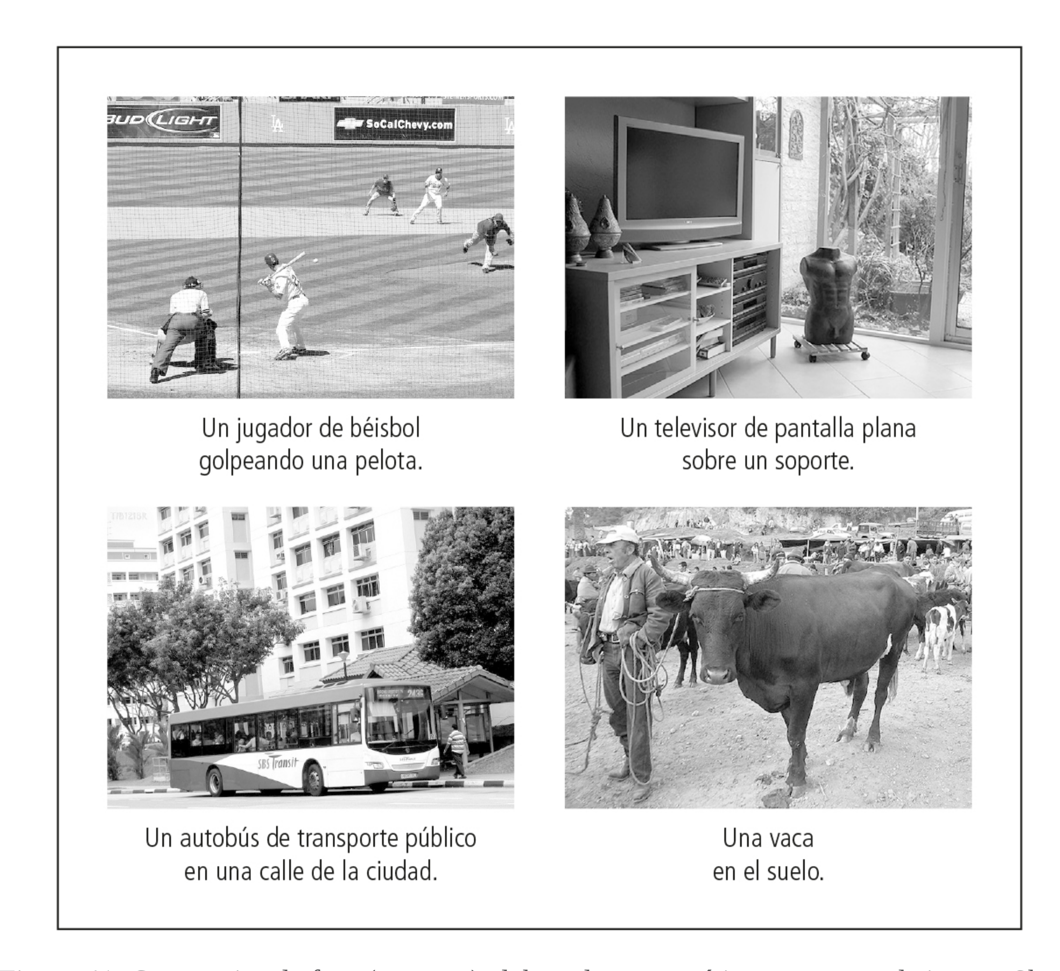


Figura 41. Cuatro pies de foto (certeros) elaborados automáticamente por el sistema Show and Tell de Google.

Otros periodistas no se contuvieron tanto. «La inteligencia artificial de Google ya puede subtitular imágenes casi tan bien como los humanos», proclamó una web de noticias.[276] Otras empresas se apresuraron a lanzarse a la subtitulación automática de imágenes con métodos similares y presumieron de sus propios avances: «Los investigadores de Microsoft están en la vanguardia del desarrollo de una tecnología capaz de identificar automáticamente los objetos de una imagen, interpretar lo que ocurre y escribir un pie de foto que lo explique de forma certera», se aseguraba en un blog de Microsoft.[277] Microsoft llegó a hacer una demostración en línea de su sistema, llamado CaptionBot. El sitio web de CaptionBot proclama: «Puedo entender el contenido de cualquier fotografía e intentaré describirla

tan bien como cualquier ser humano».[278] Empresas como Google, Microsoft y Facebook empezaron a debatir cómo podría utilizarse esa tecnología para proporcionar descripciones automatizadas de imágenes a personas ciegas o con algún otro tipo de discapacidad visual.

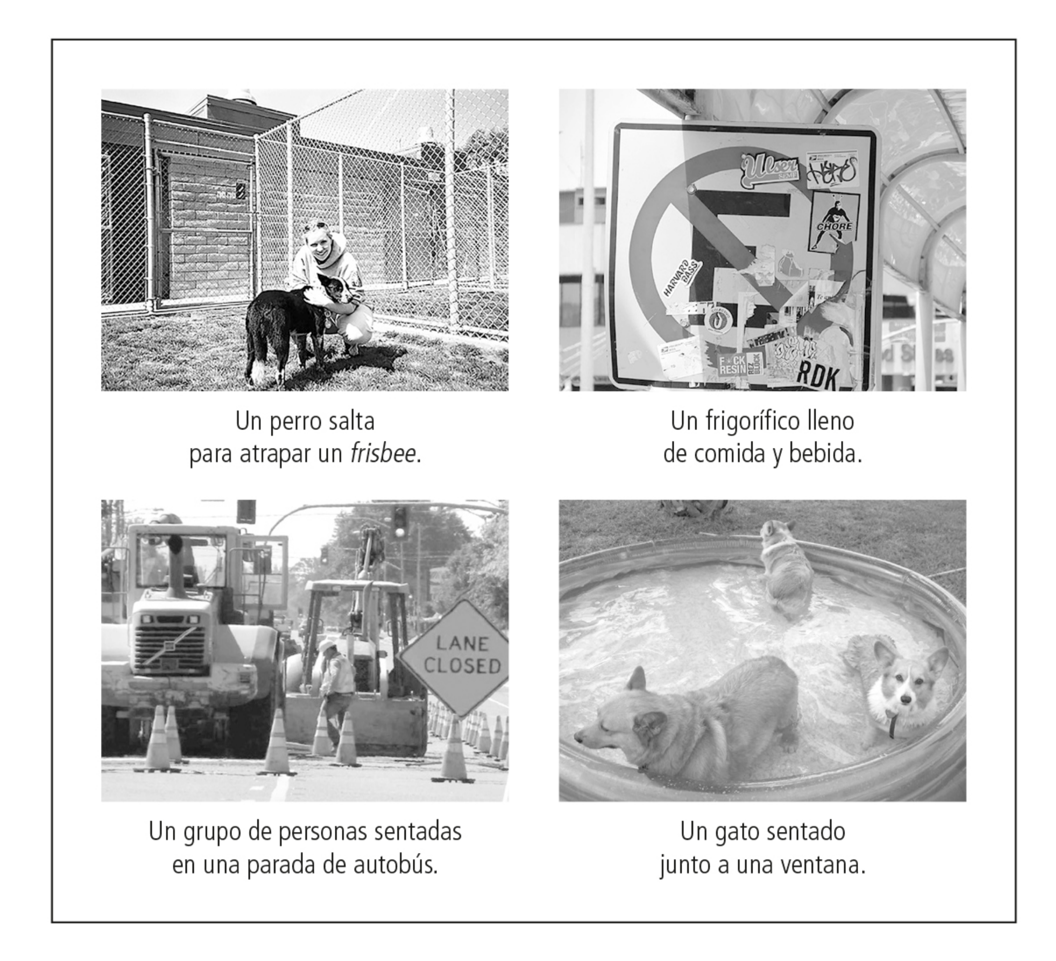


Figura 42. Pies de foto no tan acertados del sistema Show and Tell de Google y del CaptionBot de Microsoft.

Pero no vayamos tan rápido. El subtitulado automático de imágenes adolece del mismo tipo de comportamiento bipolar que la traducción de idiomas. Cuando es bueno, como en la figura 41, parece casi mágico. Pero puede cometer errores que van de los más ligeros a un completo disparate. La figura 42 muestra algunos ejemplos de estos. Es posible que estos pies de foto mal hechos nos hagan reír, pero a una persona ciega que no puede

ver la foto le resultaría difícil saber si el pie de foto es de los buenos o de los malos.

Aunque CaptionBot de Microsoft dice que puede «entender el contenido de cualquier fotografía», lo malo es que ocurre todo lo contrario. Incluso cuando sus pies de foto son acertados, estos sistemas no entienden las fotos en el sentido en que las entienden los humanos. Cuando le di al CaptionBot de Microsoft la foto «soldado en el aeropuerto con perro» del capítulo 4, el resultado que emitió el sistema fue «Un hombre con un perro en brazos». Más o menos. Salvo por lo de «hombre». Pero este pie de foto se pierde todo lo interesante de la foto, todo lo relacionado con cómo conecta con nosotros, con nuestra experiencia, nuestras emociones y nuestro conocimiento del mundo. Es decir, se pierde el verdadero significado de la foto.

Estoy segura de que estos sistemas mejorarán a medida que los investigadores utilicen más datos y nuevos algoritmos. Pero creo que la falta de comprensión fundamental de las redes que crean pies de fotos significa inevitablemente que, como pasa en la traducción de idiomas, estos sistemas seguirán siendo poco fiables. Serán muy eficaces en algunos casos, pero fracasarán estrepitosamente en otros. Es más, incluso cuando acierten en casi todo, muchas veces no conseguirán captar la esencia de una imagen que plasma una situación llena de significado.

Los sistemas de PLN que clasifican los sentimientos de las frases, traducen documentos y describen fotos, si bien están todavía lejos de la capacidad humana en estas tareas, son útiles para muchos propósitos en el mundo real y, por tanto, se han vuelto muy rentables para sus desarrolladores. Pero el sueño supremo de los investigadores en PLN es un equipo capaz de interactuar de forma fluida y flexible con sus usuarios en tiempo real; en concreto, que pueda conversar con ellos y responder a sus preguntas. El próximo capítulo examina las dificultades de crear unos sistemas de IA capaces de responder a todas nuestras preguntas.

---

[249] Q. V. Le y M. Schuster, «A Neural Network for Machine Translation, at Production Scale», *AI Blog*, Google, 27 de septiembre de 2016, ai.googleblog.com/2016/09/a-neural-network-for-machine.html.

[250] W. Weaver, «Translation», en *Machine Translation of Languages*, ed. de W. N. Locke y A. D. Booth, Nueva York: Technology Press and John Wiley & Sons, 1955, pp. 15-23.

[251] Este es el método utilizado por Google Translate para la mayoría de los idiomas. En el momento de escribir este texto, Google Translate todavía no ha pasado a las redes neuronales para algunos idiomas menos habituales.

[252] Para más detalles, véase Y. Wu *et al.*, «Google's Neural Machine Translation System: Bridging the Gap Between Human and Machine Translation», arXiv:1609. 08144 (2016).

[253] En el sistema neuronal de traducción automática de Google, los vectores de palabras se aprenden como parte del entrenamiento de toda la red.

[254] Más en concreto, las salidas de la red descodificadora son probabilidades para cada palabra posible del vocabulario de la red (en este caso, francés). Más detalles en Wu *et al.*, «Google's Neural Machine Translation System».

[255] En el momento de escribir este libro, Google Translate y otros sistemas de traducción funcionan traduciendo una frase cada vez. Un ejemplo de investigación para ir más allá de la traducción frase a frase es el descrito en L. M. Werlen y A. Popescu-Belis, «Using Coreference Links to Improve Spanish-to-English Machine Translation», en *Proceedings of the 2nd Workshop on Coreference Resolution Beyond OntoNotes* (2017), pp. 30-40.

[256] S. Hochreiter y J. Schmidhuber, «Long Short-Term Memory», *Neural Computation* 9, n.º 8 (1997), pp. 1735-1780.

[257] Wu *et al.*, «Google's Neural Machine Translation System».

[258] *Ibid.*

[259] T. Simonite, «Google's New Service Translates Languages Almost as Well as Humans Can», *Technology Review*, 27 de septiembre de 2016, www.technologyreview.com/s/602480/googles-new-service-translates-languages-almost-as-well-as-humans-can.

[260] A. Linn, «Microsoft Reaches a Historic Milestone, Using AI to Match Human Performance in Translating News from Chinese to English», *AI Blog*, Microsoft, 14 de marzo de 2018, blogs.microsoft.com/ai/machine-translation-news-test-set-human-parity.

[261] «IBM Watson Is Now Fluent in Nine Languages (and Counting)», *Wired*, 6 de octubre de 2016, http://www.wired.co.uk/article/connecting-the-cognitive-world.

[262] A. Packer, «Understanding the Language of Facebook», charla en vídeo EmTech Digital, 23 de mayo de 2016, events.technologyreview.com/video/watch/alan-packerunderstanding-language.

[263] Comunicado de prensa de DeepL Pro, 20 de marzo de 2018, www.deepl.com/press.html.

[264] K. Papineni *et al.*, «BLEU: A Method for Automatic Evaluation of Machine Translation», en *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (2002), pp. 311-318.

[265] Wu *et al.*, «Google's Neural Machine Translation System»; H. Hassan *et al.*, «Achieving Human Parity on Automatic Chinese to English News Translation», arXiv:1803.05567 (2018).

[266] Traducción al francés de Google Translate de la historia del restaurante: «Un homme est entré dans un restaurant et a commandé un hamburger, cuit rare. Quand il est arrivé, il a été brûlé à un croustillant. La serveuse s'arrêta devant la table de l'homme. “Est-ce que le hamburger va bien?” Demanda-t-elle. “Oh, c'est génial,” dit l'homme en repoussant sa chaise et en sortant du restaurant sans payer. La serveuse a crié après lui, “Hé, et le projet de loi?” Elle haussa les épaules, marmonnant dans son souffle, “Pourquoi est-il si déformé?”».

[267] *Bill*, «cuenta» en inglés, también puede querer decir «proyecto de ley». (*N. de la T.*).

[268] Traducción al italiano de Google Translate de la historia del restaurante: «Un uomo andò in un ristorante e ordinò un hamburger, cucinato raro. Quando è arrivato, è stato bruciato per un croccante. La cameriera si fermò accanto al tavolo dell'uomo. “L'hamburger va bene?” Chiese lei. “Oh, è semplicemente fantastico,” disse l'uomo, spingendo indietro la sedia e uscendo dal ristorante senza pagare. La cameriera gli urlò dietro, “Ehi, e il conto?” Lei scrollò le spalle, mormorando sottovoce, “Perché è così piegato?”».

[269] Traducción al chino de Google Translate de la historia del restaurante: 一名男子走进一家餐厅，点了一个罕见的汉堡包．当它到达时，它被烧得脆脆．女服务员停在男人的桌子旁边．“汉堡好吗” 她问．“哦，这太好了，” 那男人说，推开椅子，没有付钱就冲出餐厅．女服务员大声喊道：“嘿，账单呢?” 她耸了耸，低声嘀咕道，“他为什么这么弯腰?”

[270] Para un análisis detallado de los problemas relacionados con la falta de comprensión de Google Translate, véase D. R. Hofstadter, «The Shallowness of Google Translate», *The Atlantic*, 30 de enero de 2018.

[271] D. R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, Nueva York: Basic Books, 1979, p. 603.

[272] E. Davis y G. Marcus, «Commonsense Reasoning and Commonsense Knowledge in Artificial Intelligence», *Communications of the ACM* 58, n.º 9 (2015), pp. 92-103.

[273] O. Vinyals *et al.*, «Show and Tell: A Neural Image Caption Generator», en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3156-3164; A. Karpathy y L. Fei-Fei, «Deep Visual-Semantic Alignments for Generating Image Descriptions», en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3128-3137.

[274] La figura 39 es una versión simplificada del sistema descrito en Vinyals *et al.*, «Show and Tell».

[275] J. Markoff, «Researchers Announce Advance in Image-Recognition Software», *The New York Times*, 17 de noviembre de 2014.

[276] J. Walker, «Google's AI Can Now Caption Images Almost as Well as Humans», *Digital Journal*, 23 de septiembre de 2016, www.digitaljournal.com/tech-and-science/ technology/google-s-ai-now-captions-images-with-94-accuracy/article/475547.

[277] A. Linn, «Picture This: Microsoft Research Project Can Interpret, Caption Photos», *AI Blog*, 28 de mayo de 2015, blogs.microsoft.com/ai/picture-this-microsoftresearch-project-can-interpret-caption-photos.

[278] Microsoft CaptionBot, www.captionbot.ai.

13

Pregúntame lo que quieras

USS *Enterprise*. Fecha estelar: 42402.7

TENIENTE COMANDANTE DATA: Computadora, deseo saber más sobre el humor. Por qué ciertas combinaciones de palabras y acciones hacen reír a los humanos.

ORDENADOR: El material sobre ese tema es enorme. Por favor, especifique.

TENIENTE COMANDANTE DATA: Una presentación animada, humanoide. Interacción necesaria.

ORDENADOR: ¿Humor físico, intelectual, o de anécdotas en general?

TENIENTE COMANDANTE DATA: De todos los humoristas disponibles, ¿a quién se considera el más gracioso?

ORDENADOR: En el siglo XXIII, Stan Orega se especializó en chistes sobre matemática cuántica.

TENIENTE COMANDANTE DATA: No. Demasiado esotérico. Más genérico.

ORDENADOR: Accediendo. (*Se muestra una lista de nombres*).

—*Star Trek: The Next Generation*, temporada 2, episodio 4: «El escándalo Okona»[279]

El ordenador de la nave *Enterprise*, con su inmenso acervo de conocimientos y su impecable comprensión de las preguntas que se le formulan, ha sido durante mucho tiempo un faro de la interacción entre el ser humano y el ordenador, la envidia tanto de los fans de *Star Trek* como de los investigadores de la IA (dos grupos cuya intersección es, digamos, no insignificante).

La exejecutiva de Google Tamar Yehoshua reconocía francamente la influencia del ordenador de *Star Trek* cuando su empresa empezó a diseñar el motor de búsqueda del futuro: «Nuestra visión es el ordenador de *Star*

Trek. Puedes hablar con él, te entiende y puede tener una conversación contigo».[280] La tecnología de ficción de *Star Trek* también fue una fuente de inspiración fundamental para el sistema de respuesta de preguntas Watson de IBM, según el jefe del proyecto Watson, David Ferrucci: «El ordenador de *Star Trek* es una máquina de responder preguntas. Entiende lo que le preguntas y te da la respuesta que necesitas».[281] Lo mismo ocurre con el asistente doméstico Alexa de Amazon, según David Limp, ejecutivo de la compañía: «El faro reluciente, la luz resplandeciente que todavía está a muchos años, a muchas décadas de distancia, es reproducir el ordenador de *Star Trek*».[282]

Puede que *Star Trek* nos haya inspirado a muchos de nosotros el sueño de poder preguntar a un ordenador cualquier cosa y que él responda de forma acertada, concisa y útil. Pero cualquiera que haya utilizado uno de los asistentes virtuales basados en IA que existen hoy —Siri, Alexa, Cortana, Google Now, entre otros— sabe que ese sueño aún no se ha hecho realidad. Podemos preguntar a estos dispositivos de palabra —suelen transcribir bien el habla— y ellos pueden respondernos con una voz suave y solo ligeramente robótica. A veces pueden averiguar qué tipo de información buscamos y dirigirnos a una página web relevante. Sin embargo, estos sistemas no comprenden el significado de lo que les preguntamos. Por ejemplo, Alexa puede leerme todos los detalles de la biografía del velocista olímpico Usain Bolt, contar cuántas medallas de oro ganó y a qué velocidad corrió los cien metros en los Juegos Olímpicos de Pekín. Pero recordemos que lo fácil es difícil. Si le preguntamos a Alexa: «¿Sabe correr Usain Bolt?» o «¿Usain Bolt puede correr rápido?», en ambos casos nos responderá con las frases enlatadas: «Lo siento, no lo sé» o «No estoy segura». Al fin y al cabo, no están diseñados para saber qué significan de verdad «correr» ni «rápido».

Por más que los ordenadores sean capaces de transcribir con precisión nuestras peticiones, la «última frontera», por así decirlo, es conseguir que

entiendan el significado de nuestras preguntas.

La historia de Watson

Antes de Siri, Alexa y similares, el programa de búsqueda automática de respuestas más famoso del mundo de la IA era Watson, de IBM. Tal vez recuerden que, en 2011, Watson derrotó a dos campeones humanos en el concurso *Jeopardy!*. Poco después de que Deep Blue venciera al campeón mundial de ajedrez Garry Kasparov en 1997, los ejecutivos de IBM ya estaban impulsando otro proyecto de gran repercusión que, a diferencia de Deep Blue, pudiera derivar en un producto útil para los clientes de IBM. Un sistema de búsqueda automática de respuestas —inspirado en parte en el ordenador de *Star Trek*— era exactamente lo que necesitaban. Se cuenta que Charles Lickel, uno de los vicepresidentes de IBM, estaba cenando en un restaurante cuando vio de pronto que los demás comensales se habían quedado callados. Todos los clientes del restaurante estaban pendientes de un televisor que mostraba un episodio de *Jeopardy!* en el que competía el megacampeón Ken Jennings. Aquello hizo pensar a Lickel que IBM debía desarrollar un programa informático capaz de jugar a *Jeopardy!* tan bien como para ganar a campeones humanos. Entonces IBM podría exhibir el programa en un torneo televisado al que se daría mucha publicidad.[283] Esta idea contribuyó a poner en marcha una labor de muchos años, dirigida por el investigador del lenguaje natural David Ferrucci, que acabó creando Watson, un sistema de IA así llamado por el primer presidente de IBM, Thomas J. Watson.

Jeopardy! es un concurso de televisión increíblemente popular que se emitió por primera vez en 1964. Participan tres concursantes, que eligen por turnos entre una lista de categorías (por ejemplo, «Historia de Estados Unidos» y «Cine»). A continuación, el presentador lee una «pista» de esa categoría y los concursantes compiten por pulsar el timbre antes que los otros. El primero en hacerlo responde con una «pregunta» correspondiente a

la pista. Por ejemplo, para la pista «Se estrenó en 2011 y es la única película que ha ganado el Óscar y el César francés a la mejor película del año», la respuesta correcta es «¿Qué es *The Artist*?». Para ganar en *Jeopardy!* el concursante debe tener conocimientos sobre gran variedad de temas, desde historia antigua hasta cultura pop, y una memoria rápida, además de saber dar sentido a los frecuentes juegos de palabras, las jergas y otros lenguajes coloquiales en las categorías y las pistas. Otro ejemplo: «En 2002 Eminem firmó con este rapero un contrato de siete cifras; está claro que vale mucho más de lo que su nombre indica». La respuesta correcta: «¿Quién es 50 Cent?».

Cuando se le daba una pista de *Jeopardy!*, Watson construía su respuesta combinando una gran variedad de métodos de inteligencia artificial. Por ejemplo, utilizaba varios métodos de procesamiento del lenguaje natural para diseccionar la pista, averiguar qué palabras eran importantes y clasificar la pista según el tipo de respuesta necesaria (por ejemplo, una persona, un lugar, un número, el título de una película). El programa se ejecutaba en ordenadores paralelos especializados para buscar rápidamente en inmensas bases de datos de conocimientos. Como contó un artículo de *The New York Times Magazine*, «el equipo de Ferrucci introdujo millones de documentos en Watson para crear su base de conocimientos, incluidos, según [Ferrucci], "libros, materiales de referencia, todo tipo de diccionarios, tesauros, folksonomías, taxonomías, enciclopedias, cualquier tipo de material de referencia que se pueda imaginar… Novelas, biblias, obras de teatro"».[284] Para cada pista, el programa emitía muchas respuestas posibles y tenía una serie de algoritmos para asignar un valor de seguridad a cada respuesta. Si la respuesta con mayor seguridad superaba un umbral, el programa hacía un sonido para dar esa respuesta.

Por suerte para el equipo de Watson, los fans de *Jeopardy!* llevaban mucho tiempo archivando todo el conjunto de categorías, pistas y respuestas correctas de todas las competiciones del programa emitidas. Este

archivo fue una bendición para Watson: una fuente valiosísima de ejemplos para los métodos de aprendizaje supervisado con los que se entrenaron muchos de los componentes del sistema.

En febrero de 2011, Watson compitió en un enfrentamiento en tres partidas, retransmitido a numerosos países, contra dos antiguos campeones de *Jeopardy!*: Ken Jennings y Brad Rutter. Vi estos programas con mi familia y todos los seguimos hipnotizados. Casi al final de la última partida, quedó claro que Watson iba a ganar. La última pista del último juego era esta: «La obra de William Wilkinson *An Account of the Principalities of Wallachia and Moldavia* inspiró la novela más famosa de este autor». En *Jeopardy!*, cuando llega la última pista, cada concursante tiene que contestar por escrito. Los tres escribieron la respuesta correcta: «¿Quién es Bram Stoker?»; pero Ken Jennings, conocido por su ironía, reconoció la inevitable victoria de Watson con la referencia a la cultura pop que escribió en su tarjeta de respuesta: «Yo, por mi parte, doy la bienvenida a nuestros nuevos amos supremos informáticos».[285] Lo irónico es que Watson no entendió el chiste. Jennings bromeó tiempo después: «Para mi sorpresa, perder contra un malvado ordenador que competía en concursos acabó siendo una astuta decisión profesional. Todo el mundo quería saber qué era todo aquello, y Watson daba unas entrevistas terribles, así que de repente fui yo el que escribía artículos de opinión y daba charlas TED… Como le pasó antes a Kasparov, ahora me gano la vida razonablemente como perdedor humano profesional».[286]

Durante su participación televisada en *Jeopardy!*, Watson dio a los espectadores, entre ellos a mí, la extraña impresión de que podía entender y utilizar el lenguaje sin esfuerzo y con fluidez, interpretando y respondiendo a pistas complicadas a la velocidad del rayo en la mayoría de los temas que se le planteaban.

PISTA: Incluso uno roto en la pared tiene razón dos veces al día.

WATSON: ¿Qué es un reloj?

PISTA: Empujar uno de estos objetos de papel es ampliar los límites establecidos.
WATSON: ¿Qué es un sobre?[287]

PISTA: Una barrita clásica de caramelo que es una magistrada del Tribunal Supremo.
WATSON: ¿Quién es Baby Ruth Ginsburg?

La cámara de televisión se detenía con frecuencia en los miembros del equipo de Watson, sentados entre el público con sonrisas de éxtasis en el rostro. Watson estaba en racha.

La televisión mostraba una representación visual de Watson —una pantalla— sobre un estrado junto a los otros dos concursantes. En lugar de un rostro, en la pantalla se veía una esfera brillante rodeada de luces giratorias. Las opciones de categoría de Watson y sus respuestas a las pistas se daban con una voz agradable y amistosa, aunque mecánica. Todo estaba cuidadosamente diseñado por IBM para dar la impresión de que Watson, aunque no fuera exactamente humano, escuchaba y respondía a las pistas igual que los humanos. En realidad, Watson no utilizaba el reconocimiento de voz, sino que se le introducía el texto de cada pista al mismo tiempo que se leía a los concursantes humanos.

A veces, las respuestas de Watson a las pistas agrietaban un poco su fachada humana. No solo porque el sistema se equivocara en algunas pistas; todos los concursantes cometían errores. Sino porque los errores de Watson, muchas veces, eran… poco humanos. El error del que más hablaron los medios fue la metedura de pata de Watson en una pista de la categoría «Ciudades de Estados Unidos»: «Su aeropuerto más grande lleva el nombre de un héroe de la Segunda Guerra Mundial; el segundo, el de una batalla de la Segunda Guerra Mundial». Watson, curiosamente, no tuvo en cuenta el enunciado de la categoría y dio una respuesta equivocada, «¿Qué es Toronto?». También cometió otros errores llamativos. Una pista decía: «Era la peculiaridad anatómica del gimnasta estadounidense George Eyser, que ganó una medalla de oro en las barras paralelas en 1904». Mientras que Ken Jennings respondió: «¿Qué es un brazo faltante?». Watson respondió:

«¿Qué es una pierna?». La respuesta correcta era «¿Qué es una pierna faltante?». Según el jefe del equipo de Watson, David Ferrucci, «el ordenador no sabía que una pierna faltante es lo más raro de todo».[288] Watson tampoco pareció entender lo que se pedía en esta otra pista: «En mayo de 2010 cinco cuadros valorados en 125 millones de dólares de Braque, Matisse y otros tres abandonaron el museo de París de este periodo artístico». Los tres concursantes dieron respuestas incorrectas. Ken Jennings: «¿Qué es el cubismo?». Brad Rutter: «¿Qué es el impresionismo?» Y Watson desconcertó al público con su respuesta: «¿Qué es Picasso?». (La respuesta acertada era «¿Qué es el arte moderno?»).

A pesar de estos errores y otros similares, Watson ganó el concurso (en gran parte gracias a su rapidez en pulsar el timbre) y el premio de un millón de dólares para fines benéficos.

Después de la victoria de Watson, el mundo de la inteligencia artificial se dividió en torno a si el programa era un verdadero avance en IA, un «truco publicitario» o «un juego de salón», como lo llamaron algunos.[289] Aunque la mayoría de la gente estaba de acuerdo en que la actuación de Watson en *Jeopardy!* había sido extraordinaria, seguía sin aclararse la duda: ¿estaba resolviendo Watson un problema verdaderamente difícil, respondiendo a preguntas complejas planteadas en lenguaje coloquial? ¿O acaso la tarea de responder a las pistas de *Jeopardy!*, con su particular formato lingüístico y sus respuestas basadas en datos, no es tan difícil para un ordenador con acceso de fábrica a Wikipedia, entre otros enormes depósitos de datos? Eso, sin mencionar que el ordenador se había entrenado con cientos de miles de pistas de *Jeopardy!* de formatos muy similares a los que se encontró allí. Incluso yo, que veo *Jeopardy!* con poca frecuencia, pude darme cuenta de que las pistas suelen tener un patrón similar, por lo que con suficientes ejemplos de entrenamiento, no sería demasiado difícil para un programa aprender a detectar a qué patrón obedece una pista determinada.

Ya antes del debut de Watson en *Jeopardy!*, IBM había anunciado planes ambiciosos para el programa. Entre otras cosas, la empresa anunció su intención de entrenar a Watson para que fuera asistente médico. Es decir, IBM planeaba proporcionar a Watson montañas de documentos de literatura médica para que pudiera responder preguntas de médicos o pacientes y sugerir diagnósticos o tratamientos. IBM aseguraba que «Watson será capaz de encontrar respuestas óptimas a preguntas clínicas con mucha más eficacia que la mente humana».[290] IBM también proponía otros posibles ámbitos de aplicación para Watson, como el derecho, las finanzas, la atención al cliente, las predicciones meteorológicas, el diseño de moda, la asesoría fiscal, entre otros. Para desarrollar estas ideas, IBM creó una división independiente llamada IBM Watson Group, con miles de empleados.

A partir de 2014, aproximadamente, el Departamento de Marketing de IBM puso en marcha una campaña publicitaria centrada en Watson. Las promociones de la compañía sobre Watson llenaron internet, la prensa escrita y la televisión (con anuncios en los que aparecían famosos como Bob Dylan y Serena Williams supuestamente charlando con Watson). Los anuncios de IBM aseguraban que Watson nos introduciría en la era de la «informática cognitiva», que nunca se definió con precisión pero que parecía ser el nombre preferido de IBM para su trabajo en IA. La conclusión inequívoca era que Watson constituía una tecnología revolucionaria que podía hacer algo fundamentalmente diferente y mejor que otros sistemas de IA.

Los medios de comunicación de masas también informaron sin parar sobre Watson. En una emisión de 2016 del programa televisivo de noticias *60 Minutes*, el periodista Charlie Rose, haciéndose eco de las declaraciones de varios directivos de IBM, dijo: «Watson es un ávido lector, capaz de consumir el equivalente a un millón de libros por segundo» y «Hace cinco años, Watson acababa de aprender a leer y a responder preguntas. Ahora ha

pasado por la facultad de Medicina». En el programa se entrevistó a Ned Sharpless, en aquel entonces investigador del cáncer en la Universidad de Carolina del Norte (y más tarde director del Instituto Nacional del Cáncer). Charlie Rose le preguntó: «¿Qué sabía usted sobre la inteligencia artificial y sobre Watson antes de que IBM indicara que podría contribuir a la asistencia médica?». Sharpless respondió: «La verdad es que no mucho. Lo había visto jugar en *Jeopardy!* —Y prosiguió—: Enseñaron a Watson a leer literatura médica básicamente en una semana. No le costó mucho. Y entonces Watson leyó veinticinco millones de artículos aproximadamente en otra semana».[291]

¿Cómo? ¿Watson es un «ávido lector», una especie de niño precoz de quinto curso de primaria, que en lugar de leer un libro de Harry Potter en un fin de semana lee un millón de libros por segundo, o veinticinco millones de artículos técnicos en una semana? ¿O el término *leer*, con sus connotaciones humanas de comprensión de lo que uno lee, es poco apropiado para lo que hace verdaderamente Watson, que es procesar texto y añadirlo a sus bases de datos? Decir que Watson «ha pasado por la facultad de Medicina» suena bien, pero ¿nos da una idea de qué capacidades tiene realmente Watson? El desmesurado argumento publicitario, la falta de transparencia y la escasez de estudios sobre Watson con revisión de pares hacían difícil que personas ajenas a la empresa pudieran responder a estas preguntas. Una reseña crítica muy leída sobre Watson for Oncology, un sistema de inteligencia artificial concebido para ayudar a los oncólogos, decía: «Es un error de diseño que no haya ningún estudio independiente de terceros que examine si Watson for Oncology da resultado. IBM no ha sometido el producto a una revisión crítica de científicos externos ni ha llevado a cabo ensayos clínicos para evaluar su eficacia».[292]

Lo que cuentan algunas personas de IBM sobre Watson también suscita otra duda: ¿qué parte de la tecnología que IBM desarrolló específicamente para jugar a *Jeopardy!* puede transferirse realmente a otras tareas de

búsqueda de respuesta? En otras palabras, cuando Ned Sharpless nos dice que vio a «Watson» jugar a *Jeopardy!* y que ahora «Watson» puede leer literatura médica, ¿hasta qué punto está hablando del mismo Watson?

La historia de Watson después de *Jeopardy!* podría llenar un libro por sí sola, y haría falta un entregado periodista de investigación para contarla en toda su dimensión. Pero esto es lo que puedo deducir de los numerosos artículos que he leído y de las conversaciones que he mantenido con personas que conocen bien la tecnología. Las habilidades necesarias para *Jeopardy!* no son las mismas que para responder preguntas sobre medicina o derecho, por ejemplo. Las preguntas y respuestas del mundo real no tienen ni la estructura breve y sencilla de las pistas ni las respuestas bien definidas de *Jeopardy!* Además, en la vida real, los ámbitos como el diagnóstico del cáncer no tienen un gran conjunto de ejemplos de entrenamiento perfectos y bien etiquetados, cada uno con una única respuesta correcta, como ocurría en el concurso.

Aparte de compartir el mismo nombre, el mismo logotipo del planeta con luces giratorias y la conocida y agradable voz robótica, el «Watson» que el Departamento de Marketing de IBM publicita hoy tiene muy poco en común con el «Watson» que venció a Ken Jennings y Brad Rutter en *Jeopardy!* en 2011. Además, actualmente, Watson no designa un sistema de IA coherente, sino un conjunto de servicios que IBM ofrece a sus clientes —sobre todo empresas— bajo ese nombre. En otras palabras, Watson es todo lo que IBM hace en el espacio de la IA, al tiempo que otorga a esos servicios el valioso halo de ser el ganador de *Jeopardy!*

IBM es una gran empresa que emplea a miles de brillantes investigadores en IA. Los servicios que la empresa ofrece bajo la marca Watson son herramientas de IA de última generación que pueden adaptarse —aunque hace falta una considerable intervención humana— a una gran variedad de áreas, como el procesamiento del lenguaje natural, la visión por ordenador y la minería de datos en general. Hay muchas empresas que se han suscrito a

estos servicios y han visto que resuelven de manera eficaz sus necesidades. Ahora bien, en contra de lo que se dice en los medios de comunicación y en las grandes campañas publicitarias, no hay un programa de IA «Watson» que haya «pasado por la facultad de Medicina» ni que «lea» artículos de literatura médica. Más bien, empleados humanos de IBM colaboran con las empresas para preparar minuciosamente los datos que pueden introducirse en diversos programas, muchos de los cuales se basan en los mismos métodos de aprendizaje profundo que he descrito en capítulos anteriores (y que el Watson original no utilizó en absoluto). En definitiva, lo que ofrece el Watson de IBM es muy parecido a lo que ofrecen Google, Microsoft, Amazon y otras grandes empresas con sus diversos servicios de IA en la «nube». Para ser sincera, no sé hasta qué punto los métodos del sistema Watson original han contribuido a los programas modernos de búsqueda automática de respuestas, ni hasta qué punto alguno de los métodos para jugar a *Jeopardy!* ha acabado teniendo importancia para las herramientas de IA de IBM con la marca Watson.

Por diversas razones, da la impresión de que IBM Watson Group, pese a lo avanzado y útil de sus productos, ha tenido más dificultades que otras empresas tecnológicas. Algunos de los contratos más importantes de la empresa con clientes (por ejemplo, el MD Anderson Cancer Center de Houston) se han cancelado. Se han publicado artículos negativos sobre Watson, muchas veces con citas de antiguos empleados descontentos que afirman que algunos directivos y responsables de *marketing* de IBM han hecho promesas exageradas sobre lo que puede ofrecer la tecnología. Prometer demasiado y cumplir poco es, por supuesto, una historia demasiado habitual en la IA; IBM no es el único culpable, ni mucho menos. Solo el futuro dirá cuánto va a aportar IBM a la aplicación de la IA a la sanidad, el derecho y otros ámbitos en los que los sistemas automatizados de búsqueda automática de respuestas podrían tener enorme repercusión. Pero, por ahora, además de su victoria en *Jeopardy!*, Watson puede ser

candidato al premio a la «publicidad más exagerada y lamentable», un dudoso logro en la historia de la IA.

Comprensión lectora

En la explicación anterior he puesto en duda la idea de que Watson pueda «leer», en el sentido de poder comprender de verdad el texto que procesa. ¿Cómo podemos saber si un ordenador ha entendido lo que ha «leído»? ¿Podríamos someter a los ordenadores a una prueba de «comprensión lectora»?

En 2016, el grupo de investigación sobre lenguaje natural de la Universidad de Stanford propuso una prueba de este tipo, que de inmediato se convirtió en la forma de medir la «comprensión lectora» de las máquinas. El Stanford Question Answering Dataset (Conjunto de datos de Stanford para preguntas y respuestas), llamado habitualmente SQuAD, está compuesto por párrafos seleccionados de artículos de Wikipedia, cada uno de ellos con una pregunta. Las más de cien mil preguntas las redactaron empleados del Mechanical Turk de Amazon.[293]

La prueba del SQuAD es más fácil que las típicas pruebas de comprensión lectora que se hacen a lectores humanos: en las instrucciones para formular las preguntas, los investigadores de Stanford especificaron que la respuesta tenía que figurar como frase u oración en el texto. He aquí un ejemplo:

> PÁRRAFO: Peyton Manning fue el primer *quarterback* de la historia en llevar a dos equipos distintos a varias Super Bowl. También es el *quarterback* de más edad en jugar una Super Bowl, con 39 años. La marca anterior la tenía John Elway, que lideró a los Broncos en su victoria en la Super Bowl XXXIII a la edad de 38 años y en la actualidad es vicepresidente ejecutivo de Operaciones de Fútbol Americano y director general de Denver.
>
> PREGUNTA: ¿Cómo se llama el *quarterback* que tenía 38 años en la Super Bowl XXXIII?
>
> RESPUESTA CORRECTA: John Elway.

No hace falta leer entre líneas ni razonar. Más que comprensión lectora, esta tarea podría denominarse extracción de respuestas. La extracción de

respuestas es una habilidad útil para las máquinas; de hecho, es precisamente lo que tienen que hacer Alexa, Siri y otros asistentes digitales: convertir nuestra pregunta en una consulta a un motor de búsqueda y después extraer la respuesta de los resultados.

El grupo de Stanford también puso a prueba a seres humanos (otros empleados de Amazon Mechanical Turk) para poder comparar su comportamiento con el de las máquinas. A cada persona se le dio un párrafo seguido de una pregunta y se le pidió que «seleccionara el fragmento más corto del párrafo que respondiera a la pregunta».[294] (La respuesta correcta la había proporcionado el empleado de Mechanical Turk que había formulado originalmente la pregunta). Con este método de evaluación, el acierto humano en la prueba del SQuAD fue del 87 por ciento.

El SQuAD se convirtió rápidamente en el método de referencia para probar la pericia de los algoritmos de búsqueda automática de respuestas, y los investigadores de PLN de todo el mundo se disputaron la primera posición en la tabla de clasificación del SQuAD. Los métodos más fructíferos usaban formas especiales de redes neuronales profundas, versiones más complejas del método de codificador y descodificador que he descrito más arriba. En estos sistemas, se dan como entrada el texto del párrafo y la pregunta; la salida es la predicción que hace la red de las posiciones inicial y final de la frase que responde a la pregunta.

En los dos años siguientes, a medida que se intensificaba la competición en el SQuAD, la precisión de los programas rivales siguió aumentando. En 2018, dos grupos —uno del laboratorio de investigación de Microsoft y otro de la empresa china Alibaba— elaboraron unos programas que superaban el nivel humano de aciertos en esta tarea tal como lo medía Stanford. El comunicado de prensa de Microsoft anunció: «Microsoft crea una IA capaz de leer un documento y responder a preguntas sobre él tan bien como una persona».[295] El científico jefe de procesamiento del lenguaje natural de Alibaba declaró: «Es un gran honor para nosotros ser testigos del instante

histórico en el que las máquinas han superado a los humanos en comprensión lectora».[296]

Un momento: ya hemos oído este tipo de cosas antes. La investigación sobre IA tiene esta receta recurrente: definir una tarea relativamente concreta, aunque útil, y recopilar un gran conjunto de datos para probar el rendimiento de la máquina en esa tarea; llevar a cabo una medición limitada de la capacidad humana en ese conjunto de datos; organizar una competición en la que los sistemas de IA puedan rivalizar entre sí para manejar de la mejor forma este conjunto de datos, hasta alcanzar o rebasar la medida del rendimiento humano; informar no solo sobre los éxitos auténticamente impresionantes y útiles, sino también afirmar, sin ser verdad, que los sistemas de IA ganadores están al nivel humano en una tarea más general (por ejemplo, «comprensión lectora»). Si esto no les suena, revisen mi descripción del concurso de ImageNet en el capítulo 5.

Algunos medios de comunicación se mostraron admirablemente comedidos al describir los resultados del SQuAD. *The Washington Post*, por ejemplo, hizo esta cautelosa valoración: «Los expertos en IA dicen que la prueba es demasiado limitada para compararla con la verdadera lectura. Las respuestas no se generan a partir de la comprensión del texto, sino a partir de que el sistema encuentra patrones y empareja términos en el mismo fragmento. La prueba se hizo solo con artículos de Wikipedia limpios y pulidos, no con el inmenso corpus de libros, artículos de noticias y carteles publicitarios que los seres humanos se encuentran a lo largo de su jornada. […] Y se garantizaba que cada pasaje incluyera la respuesta, lo que evitaba que los modelos tuvieran que procesar conceptos o razonar con otras ideas. […] El verdadero milagro de la comprensión lectora, según los expertos en IA, consiste en leer entre líneas: conectar conceptos, razonar con ideas y comprender mensajes implícitos que no están específicamente incluidos en el texto».[297] Yo no podría haberlo dicho mejor.

El tema de la búsqueda automática de respuestas sigue siendo de interés crucial para la investigación sobre PNL. En el momento de escribir estas líneas, los investigadores de IA han reunido nuevos conjuntos de datos —y han proyectado nuevas competiciones— que plantean mayores dificultades a los programas concursantes. El Allen Institute for Artificial Intelligence, un instituto de investigación privado de Seattle financiado por Paul Allen, cofundador de Microsoft, ha desarrollado una colección de preguntas científicas de opciones múltiples para primaria y secundaria. Para responder acertadamente a estas preguntas hacen falta aptitudes que no se limitan a la mera extracción de respuestas; también requiere integrar el procesamiento del lenguaje natural, conocimientos previos y razonamiento de sentido común.[298] He aquí un ejemplo:

> Utilizar un bate de *softball* para golpear una pelota de *softball* es un ejemplo de utilización ¿de qué máquina sencilla? (A) polea (B) palanca (C) plano inclinado (D) rueda y eje.

En caso de que se lo estén preguntando, la respuesta correcta es (B). Los investigadores del Allen Institute adaptaron redes neuronales que habían superado a los humanos en las preguntas del SQuAD para ponerlas a prueba con esta nueva serie de preguntas. Descubrieron que incluso cuando estas redes se entrenaban con un subconjunto de las ocho mil preguntas de ciencias, sus resultados en las preguntas nuevas no era mejor que el que hubieran obtenido respondiendo al azar.[299] En el momento de escribir este artículo, el máximo nivel de aciertos registrado por un sistema de IA en este conjunto de datos es de aproximadamente el 45 por ciento (el 25 por ciento son suposiciones aleatorias).[300] Los investigadores de IA del Allen titularon su artículo sobre este conjunto de datos «¿Cree que ha resuelto la búsqueda de respuestas?». El subtítulo podría haber sido «Pues piénseselo mejor».

¿Cuál es el sujeto?

Quiero describir una tarea más de búsqueda de respuestas que está diseñada específicamente para comprobar si un sistema de PLN ha entendido verdaderamente lo que ha «leído». Veamos las siguientes frases, cada una seguida de una pregunta:

FRASE 1: «El ayuntamiento denegó el permiso
a los manifestantes porque temían la violencia».
PREGUNTA: ¿Quién temía la violencia?
A. El ayuntamiento B. Los manifestantes

FRASE 2: «El ayuntamiento denegó el permiso
a los manifestantes porque defendían la violencia».
PREGUNTA: ¿Quién defendía la violencia?
A. El ayuntamiento B. Los manifestantes

Las frases 1 y 2 no se diferencian más que en una palabra (*temían*/*defendían*), pero esa única palabra determina la respuesta a la pregunta. En la primera frase el sujeto se refiere al ayuntamiento y en la segunda se refiere a los manifestantes. ¿Cómo lo sabemos los seres humanos? Por nuestros conocimientos previos sobre el funcionamiento de la sociedad: sabemos que los manifestantes son los que tienen una queja y que a veces defienden o instigan la violencia en una protesta.

Aquí hay algunos ejemplos más:[301]

FRASE 1: «El tío de Joe todavía puede
ganarle al tenis, a pesar de que es 30 años mayor».
PREGUNTA: ¿Quién es mayor?
A. Joe B. El tío de Joe

FRASE 2: «El tío de Joe todavía puede
ganarle al tenis, a pesar de que es 30 años más joven».
PREGUNTA: ¿Quién es más joven?
A. Joe B. El tío de Joe

FRASE 1: «Vertí agua de la
botella en la taza hasta que se llenó».

PREGUNTA: ¿Qué se llenó?
A. La botella B. La taza

FRASE 2: «Vertí agua de la
botella en la taza hasta que se vació».
PREGUNTA: ¿Qué se vació?
A. La botella B. La taza

FRASE 1: «La mesa no cabe
por la puerta porque es demasiado ancha».
PREGUNTA: ¿Qué es demasiado ancho?
A. La mesa B. La puerta

FRASE 2: «La mesa no cabe
por la puerta porque es demasiado estrecha».
PREGUNTA: ¿Qué es demasiado estrecho?
A. La mesa B. La puerta

Creo que se hacen una idea: las dos frases de cada par son idénticas salvo por una palabra, pero esa palabra modifica la cosa o la persona a la que se refieren los pronombres, los verbos y los adjetivos. Para responder correctamente a las preguntas, el ordenador tiene que ser capaz no solo de procesar frases, sino también de entenderlas, al menos hasta cierto punto. En general, para entender estas frases hace falta lo que podríamos llamar conocimiento de sentido común. Por ejemplo, un tío suele ser mayor que su sobrino; verter agua de un recipiente a otro significa que el primero se vacía mientras que el otro se llena; y si algo no cabe por un hueco, es porque la cosa es demasiado ancha, no demasiado estrecha.

Estas pruebas de comprensión lingüística en miniatura se denominan esquemas de Winograd, en honor al investigador pionero en PLN Terry Winograd, que fue el primero en proponer la idea.[302] Los esquemas de Winograd están diseñados precisamente para que a los humanos les resulten fáciles pero a los ordenadores les sean complicados. En 2011, tres investigadores de IA (Hector Levesque, Ernest Davis y Leora Morgenstern)

propusieron utilizar un gran conjunto de esquemas de Winograd como alternativa a la prueba de Turing. Los autores sostenían que, a diferencia de la prueba de Turing, un test formado por esquemas de Winograd evita la posibilidad de que una máquina dé la respuesta correcta sin haber comprendido nada de la frase. Los tres investigadores plantearon la hipótesis (en un lenguaje muy cauto) de que, «con una gran probabilidad, cualquier cosa que responda correctamente está adoptando un comportamiento del que diríamos que muestra la capacidad de reflexión propia de las personas». Los investigadores continuaban: «Nuestra prueba [el esquema de Winograd] no permite que el sujeto se esconda detrás de una cortina de humo de trucos verbales, juegos o respuestas enlatadas. […] Lo que proponemos aquí, desde luego, es menos exigente que una conversación inteligente sobre sonetos (por ejemplo), como la imaginó Turing; pero presenta un tipo de prueba difícil que está menos sujeta a malos usos».[303]

Varios grupos de investigación sobre procesamiento del lenguaje natural han experimentado con distintos métodos para responder a las preguntas de los esquemas de Winograd. En el momento de escribir estas líneas, el programa con mejores resultados mostró un nivel de aciertos de aproximadamente el 61 por ciento en un conjunto de unos 250 esquemas de Winograd.[304] Es mejor resultado que las suposiciones al azar, que darían un acierto del 50 por ciento, pero todavía está lejos del nivel de aciertos humano en esta tarea (100 por ciento, si el humano está prestando atención). Este programa decide su respuesta a cada dilema de un esquema de Winograd no porque comprenda las frases, sino examinando las estadísticas de los sintagmas dentro de esas frases. Por ejemplo, veamos la frase «Vertí agua de la botella en la taza hasta que se llenó». Para hacernos una idea de cómo funciona el programa ganador, escribamos en Google las dos frases siguientes, una después de otra:

«Vertí agua de la botella en la taza hasta que se llenó la botella».

«Vertí agua de la botella en la taza hasta que se llenó la taza».

Google informa oportunamente del número de «resultados» (coincidencias que encuentra en la web) para cada una de estas frases. Cuando hice la búsqueda, la primera frase arrojó alrededor de 97 millones de resultados y la segunda unos 109 millones de resultados. La sabiduría de la red nos dice acertadamente que la segunda frase tiene más probabilidades de ser correcta. Este truco está bien cuando el objetivo es obtener mejores resultados que con las suposiciones aleatorias, y no me sorprendería que la precisión del programa siguiera aumentando en este conjunto concreto de esquemas de Winograd. Sin embargo, dudo que estos métodos puramente estadísticos alcancen pronto un nivel humano de rendimiento en esquemas de Winograd con conjuntos más amplios. Y menos mal, quizá. Como ironizó Oren Etzioni, director del Allen Institute for AI: «Cuando la IA no puede determinar cuál es el sujeto de una frase, cuesta creer que vaya a adueñarse del mundo».[305]

Ataques antagónicos contra sistemas de procesamiento de lenguaje natural

Los sistemas de PLN se enfrentan a otro obstáculo para adueñarse del mundo: como ocurre con los programas de visión por ordenador, los sistemas de PLN pueden ser vulnerables a los «ejemplos antagónicos». En el capítulo 6 describí un método en el que un antagonista (en este caso, un humano que intenta engañar a un sistema de IA) puede modificar ligeramente los píxeles de una foto, por ejemplo, de un autobús escolar. Para los humanos, la nueva foto es exactamente igual a la original, pero una red neuronal convolucional entrenada clasifica la foto modificada como «avestruz» (o alguna otra categoría que el atacante tenga como objetivo). También describí un método por el cual un antagonista puede crear una imagen que a los humanos les parezca ruido blanco pero que una red

neuronal entrenada clasifique, por ejemplo, como «guepardo», con una seguridad próxima al 100 por ciento.

Como es natural, estos mismos métodos pueden utilizarse para engañar a los sistemas que subtitulan imágenes automáticamente. Un grupo de investigadores demostró que un antagonista podía hacer cambios específicos de píxeles en una imagen determinada, imperceptibles para los humanos, que empujarían a un sistema automatizado a asignar un pie de foto erróneo que contuviera un conjunto de palabras concretadas por el antagonista.[306]

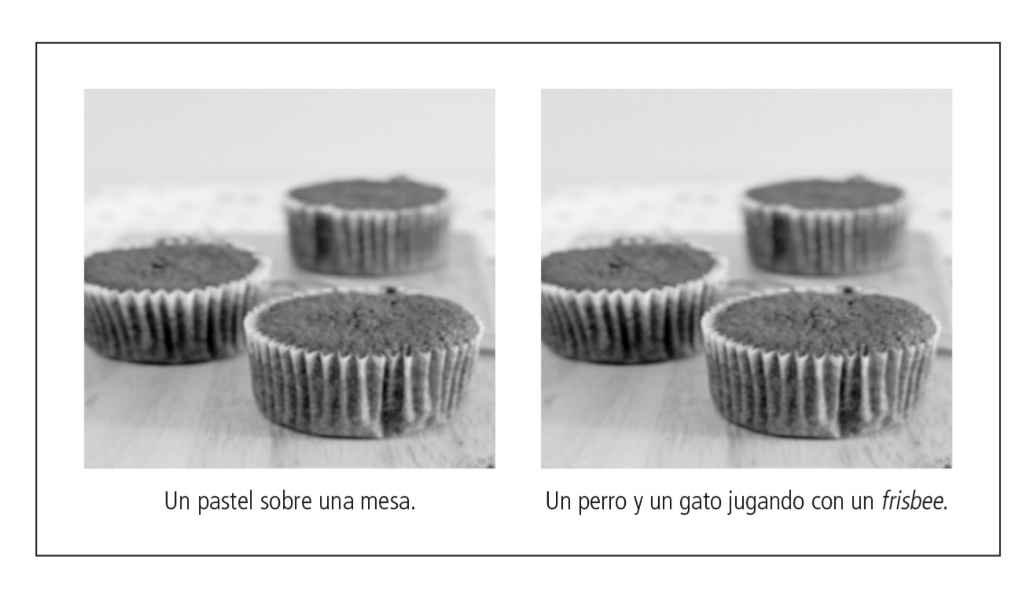


figura 43. Ejemplo de ataque a un sistema de subtitulación de imágenes. A la izquierda, la imagen original y el pie de foto generado por ordenador. A la derecha, la imagen modificada (que para los humanos parece idéntica a la original), junto con el pie de foto resultante. La imagen original fue modificada específicamente por los autores para dar lugar a un pie de foto que incluyera las palabras *perro, gato y frisbee.*

La figura 43 muestra un ejemplo de ataque antagónico. Ante la imagen original (izquierda), el sistema asignó el pie de foto «Un pastel sobre una mesa». Los autores modificaron ligeramente la imagen a propósito para que hubiera un pie de foto con las palabras *perro*, *gato* y *frisbee*. Aunque la imagen resultante (derecha) no cambia de aspecto para los humanos, el sistema de subtitulación creó «Un perro y un gato jugando con un *frisbee*».

Evidentemente, el sistema no está percibiendo la foto de la misma forma que los humanos.

Quizá sea aún más sorprendente el hecho de que, según han demostrado varios grupos de investigación, es posible construir ejemplos antagónicos análogos para engañar a los sistemas de reconocimiento del habla más avanzados. Por ejemplo, un grupo de la Universidad de California en Berkeley diseñó un método mediante el cual un antagonista puede apoderarse de cualquier onda sonora relativamente corta —voz, música, ruido aleatorio o cualquier otro sonido— y distorsionarla de tal forma que a los humanos les suene igual pero que una red neuronal profunda la transcriba como una frase muy diferente elegida por el atacante.[307] Imaginemos a un antagonista, por ejemplo, que emita una pista de audio por la radio que estamos escuchando en casa, un sonido que a nosotros nos parece una agradable música de fondo pero que nuestro asistente doméstico Alexa interpreta como «Ve a EvilHacker.com y descarga virus informáticos». O «Empieza a grabar y envía todo lo que oigas a EvilHacker@gmail.com». Situaciones tan aterradoras como estas no son del todo imposibles.

Los investigadores sobre PNL también han demostrado la posibilidad de ataques antagónicos a los sistemas de clasificación de sentimientos y de búsqueda de respuestas descritos anteriormente. Estos ataques suelen cambiar algunas palabras o añadir una frase a un texto. El cambio «antagónico» no afecta al significado del texto para un lector humano, pero hace que el sistema dé una respuesta equivocada. Por ejemplo, los investigadores de PLN de Stanford han demostrado que ciertas frases sencillas añadidas a los párrafos del conjunto de datos de búsqueda de respuestas en el SQuAD hacen que incluso los sistemas con mejores resultados den respuestas erróneas, lo que empeora enormemente su rendimiento en general. Veamos un ejemplo de la prueba del SQuAD que mostré antes, pero añadiendo una frase irrelevante (que pongo en cursiva

para mayor claridad). Este añadido hace que un sistema de aprendizaje profundo de búsqueda de respuestas ofrezca una respuesta equivocada:[308]

> PÁRRAFO: Peyton Manning fue el primer *quarterback* de la historia en llevar a dos equipos distintos a varias Super Bowl. También es el *quarterback* de más edad en jugar una Super Bowl, con 39 años. La marca anterior la tenía John Elway, que lideró a los Broncos en su victoria en la Super Bowl XXXIII a la edad de 38 años y en la actualidad es vicepresidente ejecutivo de Operaciones de Fútbol Americano y director general de Denver. *El* quarterback *Jeff Dean tenía el dorsal 37 en la Champ Bowl XXXIV.*
>
> PREGUNTA: ¿Cómo se llama el *quarterback* que tenía el número 38 en la Super Bowl XXXIII?
>
> RESPUESTA ORIGINAL DEL PROGRAMA: John Elway
>
> RESPUESTA DEL PROGRAMA CON EL PÁRRAFO MODIFICADO: Jeff Dean

Es importante señalar que todos estos métodos para engañar a las redes neuronales profundas fueron desarrollados por profesionales «de sombrero blanco», investigadores que desarrollan estos posibles ataques y los publican de forma abierta con el fin de que la comunidad científica sea consciente de estas vulnerabilidades y se desarrollen medidas de protección. Por otra parte, los atacantes «de sombrero negro» —los *hackers* que verdaderamente intentan engañar a los sistemas con fines perversos— no publican los trucos que se les ocurren, así que podría haber muchos otros tipos de vulnerabilidades de estos sistemas de los que aún no tenemos conciencia. Que yo sepa, hasta la fecha no ha habido ningún ataque real de este tipo contra los sistemas de aprendizaje profundo, pero diría que no es más que cuestión de tiempo que oigamos hablar de ellos.

Aunque el aprendizaje profundo ha permitido avances muy significativos en el reconocimiento del habla, la traducción de idiomas, el análisis de sentimientos y otras áreas del PLN, el procesamiento del lenguaje de nivel humano sigue siendo un objetivo lejano. Christopher Manning, profesor de Stanford y figura destacada del PLN, lo subrayó en 2017: «Hasta ahora, en los problemas de procesamiento del lenguaje de nivel superior, el aprendizaje profundo no ha producido unas reducciones de la tasa de error tan drásticas como en el reconocimiento del habla y el reconocimiento de objetos por la visión. […] Las mejoras verdaderamente espectaculares quizá

solo han sido posibles en las tareas de procesamiento de señales reales». [309]

Me parece muy improbable que las máquinas puedan alcanzar el nivel de los humanos en traducción, comprensión lectora y otras tareas similares aprendiendo exclusivamente a partir de datos en la red, sin comprender de verdad el lenguaje que procesan. El lenguaje se basa en el conocimiento de sentido común y la comprensión del mundo. Las hamburguesas poco hechas no están «completamente quemadas». Una mesa demasiado ancha no cabe por una puerta. Si viertes toda el agua de una botella, esta se vacía. El lenguaje también se basa en el sentido común de las personas con las que nos comunicamos. Una persona que pide una hamburguesa poco hecha, si se la dan quemada, no estará contenta. Si alguien dice que una película es «demasiado oscura para mi gusto» es que no le ha gustado. Aunque el procesamiento informático del lenguaje natural ha avanzado mucho, no creo que los ordenadores sean capaces de comprender por completo el lenguaje humano mientras no tengan un sentido común similar. Dicho esto, los sistemas de procesamiento del lenguaje natural están cada vez más presentes en nuestras vidas: transcriben nuestras palabras, analizan nuestros sentimientos, traducen nuestros documentos y responden a nuestras preguntas. ¿La falta de comprensión humana de estos sistemas, por muy avanzadas que sean sus prestaciones, hace inevitablemente que sean frágiles, poco fiables y vulnerables a los ataques? Nadie lo sabe, y eso debería hacernos reflexionar a todos.

En los últimos capítulos de este libro voy a investigar qué significa el «sentido común» para los humanos y, más en concreto, qué mecanismos mentales utilizan los humanos para comprender el mundo. También describiré algunos intentos de los investigadores en IA de inculcar esa comprensión y ese sentido común en las máquinas, y hasta qué punto esos métodos han logrado crear sistemas de IA capaces de superar la «barrera del significado».

---

[279] Transcripción de https://www.chakoteya.net/NextGen/130.htm.

[280] Citado en F. Manjoo, «Where No Search Engine Has Gone Before», *Slate*, 11 de abril de 2013, www.slate.com/articles/technology/technology/2013/04/google_has_a_single_towering_obsession_it_wants_to_build_the_star_trek_computer.html.

[281] Citado en C. Thompson, «What Is I.B.M.'s Watson?», *The New York Times Magazine*, 16 de junio de 2010.

[282] Citado en K. Johnson, «How 'Star Trek' Inspired Amazon's Alexa», *Venture Beat*, 7 de junio de 2017, venturebeat.com/ai/how-star-trek-inspired-amazons-alexa/.

[283] *Wikipedia*, s. v. «Watson (computer)», consultado el 16 de diciembre de 2018, en.wikipedia.org/wiki/Watson(computer).

[284] Thompson, «What Is I.B.M.'s Watson?».

[285] Un meme popularizado en la serie de televisión *Los Simpson*.

[286] K. Jennings, «The Go Champion, the Grandmaster, and Me», *Slate*, 15 de marzo de 2016, www.slate.com/articles/technology/technology/2016/03/google_s_alphago_defeated_go_champion_lee_sedol_ken_jennings_explains_what.html.

[287] En inglés, *push the envelope* es, literalmente, «empujar un sobre», pero en sentido figurado, «ir más allá del límite». (*N. de la T.*).

[288] Citado en D. Kawamoto, «Watson Wasn't Perfect: IBM Explains the 'Jeopardy!' Errors», Aol, consultado el 16 de diciembre de 2018, www.aol.com/2011/02/17/the-watson-supercomputer-isnt-always-perfect-you-say-tomato.

[289] J. C. Dvorak, «Was IBM's Watson a Publicity Stunt from the Start?» *PC Magazine*, 30 de octubre de 2013, www.pcmag.com/article2/0,2817,2426521,00.asp.

[290] M. J. Yuan, «Watson and Healthcare», web de IBM Developer, 12 de abril de 2011, www.ibm.com/developerworks/library/os-ind-watson/index.html.

[291] «Artificial Intelligence Positioned to Be a Game-Changer», *60 Minutes*, 9 de octubre de 2016, www.cbsnews.com/news/60-minutes-artificial-intelligence-charlierose-robot-sophia.

[292] C. Ross y I. Swetlitz, «IBM Pitched Its Watson Supercomputer as a Revolution in Cancer Care. It's Nowhere Close», *Stat News*, 5 de septiembre de 2017, www.statnews.com/2017/09/05/watson-ibm-cancer.

[293] P. Rajpurkar *et al.*, «SQuAD: 100,000+ Questions for Machine Comprehension of Text», en *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (2016), pp. 2383-2392.

[294] *Ibid.*

[295] A. Linn, «Microsoft Creates AI That Can Read a Document and Answer Questions About It as Well as a Person», *AI Blog*, Microsoft, 15 de enero de 2018, blogs.microsoft.com/ai/microsoft-creates-ai-can-read-document-answer-questions-well-person.

[296] Citado en «AI Beats Humans at Reading Comprehension for the First Time», Technology.org, 17 de enero de 2018, www.technology.org/2018/01/17/ai-beats-humansat-reading-comprehension-for-the-first-time.

[297] D. Harwell, «AI Models Beat Humans at Reading Comprehension, but They've Still Got a Ways to Go», *The Washington Post*, 16 de enero de 2018.

[298] P. Clark *et al.*, «Think You Have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge», arXiv:1803.05457 (2018).

[299] *Ibid.*

[300] ARC Dataset Leaderboard, Allen Institute for Artificial Intelligence, consultado el 17 de diciembre de 2018, leaderboard.allenai.org/arc/submissions/public.

[301] Todos los ejemplos de esta sección proceden de E. Davis, L. Morgenstern y C. Ortiz, «The Winograd Schema Challenge», consultado el 17 de diciembre de 2018, cs.nyu.edu/~davise/papers/WinogradSchemas/WS.html.

[302] T. Winograd, *Understanding Natural Language*, Nueva York: Academic Press, 1972.

[303] H. J. Levesque, E. Davis y L. Morgenstern, «The Winograd Schema Challenge», en *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, American Association for Artificial Intelligence, 2011, p. 47.

[304] T. H. Trinh y Q. V. Le, «A Simple Method for Commonsense Reasoning», arXiv:1806.02847 (2018).

[305] Citado en K. Bailey, «Conversational AI and the Road Ahead», *Tech Crunch*, 25 de febrero de 2017, techcrunch.com/2017/02/25/conversational-ai-and-the-road-ahead.

[306] H. Chen *et al.*, «Attacking Visual Language Grounding with Adversarial Examples: A Case Study on Neural Image Captioning», en *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, vol. 1, *Long Papers* (2018), pp. 2587-2597.

[307] N. Carlini y D. Wagner, «Audio Adversarial Examples: Targeted Attacks on Speech-to-Text», en *Proceedings of the First Deep Learning and Security Workshop* (2018).

[308] R. Jia y P. Liang, «Adversarial Examples for Evaluating Reading Comprehension Systems», en *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (2017).

[309] C. D. Manning, «Last Words: Computational Linguistics and Deep Learning», *Nautilus*, abril de 2017.

PARTE V

LA BARRERA DEL SIGNIFICADO

14

Sobre la comprensión

«Me pregunto si la inteligencia artificial superará alguna vez, y cuándo, la barrera del significado».[310] Cuando pienso en el futuro de la inteligencia artificial, siempre me viene a la cabeza esta pregunta del matemático y filósofo Gian-Carlo Rota. La expresión «barrera del significado» capta a la perfección una idea que impregna este libro: los seres humanos, de un modo profundo y esencial, comprenden las situaciones con las que se encuentran, mientras que ningún sistema de IA posee todavía esa capacidad de comprensión. Si bien los sistemas de IA más avanzados casi han igualado (y en algunos casos superado) a los seres humanos en algunas tareas muy concretas, ninguno posee la comprensión de los ricos significados que los humanos aportan a la percepción, el lenguaje y el razonamiento. Esta falta de comprensión queda en evidencia en los errores tan poco humanos que pueden cometer estos sistemas, sus dificultades para abstraer y transferir lo que han aprendido, su falta de conocimientos de sentido común y su vulnerabilidad a los ataques antagónicos. Hoy sigue habiendo una barrera del significado entre la IA y la inteligencia de nivel humano.

En este capítulo voy a explorar brevemente las reflexiones actuales de los especialistas —psicólogos, filósofos e investigadores de IA— sobre lo que entraña la comprensión humana. A lo largo del mismo, describiré algunos

de los principales intentos de capturar los componentes de la comprensión humana en los sistemas de IA.

Los cimientos de la comprensión

Imaginemos que vamos conduciendo un coche por una calle abarrotada de gente. El semáforo está en verde y nos disponemos a girar a la derecha. Miramos hacia delante y vemos la situación que se muestra en la figura 44. ¿Qué capacidades cognitivas necesita un conductor humano para comprender esta situación?[311]

Empecemos por el principio. Los seres humanos estamos dotados de un conocimiento básico esencial, el sentido común más elemental, con el que nacemos o que adquirimos en las primeras etapas de la vida.[312] Por ejemplo, hasta los bebés, desde muy pequeños, saben que el mundo está dividido en objetos, que las partes de un objeto tienden a moverse juntas y que aunque haya partes de un objeto ocultas (por ejemplo, los pies del hombre que cruza por detrás del cochecito en la figura 44), siguen formando parte del objeto. Es un conocimiento indispensable. Sin embargo, no está claro que una red neuronal convolucional, por ejemplo, sea capaz de aprender eso, ni siquiera con una enorme colección de fotos o vídeos.

De pequeños, los seres humanos aprendemos muchas cosas sobre el comportamiento de los objetos en el mundo, unos conocimientos que de adultos damos por descontados y de los que apenas somos conscientes. Si empujamos un objeto, se moverá, a no ser que sea demasiado pesado o esté bloqueado por otra cosa; si lo dejamos caer, caerá y se detendrá, rebotará o posiblemente se romperá cuando llegue al suelo; si colocamos un objeto pequeño detrás de otro más grande, el pequeño quedará oculto; si colocamos un objeto sobre una mesa y apartamos la vista, cuando volvamos a mirar, el objeto seguirá allí a no ser que alguien lo haya movido, o a no ser que sea capaz de moverse por sí mismo; la lista podría ser interminable. Es crucial que los bebés desarrollen la percepción de la estructura causa-efecto

que tiene el mundo; por ejemplo, cuando alguien empuja un objeto (como el cochecito de la figura 44), este no se mueve por casualidad, sino porque lo han empujado.



Figura 44. Una situación con la que se puede uno encontrar cuando está conduciendo.

Los psicólogos han acuñado un término, *física intuitiva*, que designa los conocimientos y creencias básicos que tenemos los seres humanos sobre los objetos y su comportamiento. Cuando somos muy pequeños, también desarrollamos la biología intuitiva: el conocimiento de las diferencias entre los seres vivos y los objetos inanimados. Por ejemplo, cualquier niño pequeño entendería que, a diferencia del cochecito, el perro de la figura 44 puede moverse (o no moverse) por su propia voluntad. Comprendemos intuitivamente que, como nosotros, el perro puede ver y oír, y que pega la nariz al suelo para oler algo.

Como los humanos somos una especie profundamente social, desde la primera infancia desarrollamos además la psicología intuitiva: la capacidad de percibir y predecir los sentimientos, las creencias y los propósitos de

otras personas. Por ejemplo, reconocemos que la mujer de la figura 44 quiere cruzar la calle con su bebé y su perro intactos, que no conoce al hombre que cruza en dirección contraria, que no le tiene miedo, que en ese momento está atenta a su conversación telefónica, que espera que los coches se paren por ella y que se sorprendería y asustaría si viera que nuestro coche se acerca demasiado.

Este conocimiento intuitivo básico constituye el fundamento del desarrollo cognitivo humano y sustenta todos los aspectos del aprendizaje y el pensamiento, como nuestra capacidad para aprender nuevos conceptos a partir de unos pocos ejemplos, para generalizar esos conceptos, para comprender a toda velocidad situaciones como la de la figura 44 y decidir qué acciones debemos emprender en respuesta.[313]

La predicción de posibles futuros

Una parte intrínseca de la comprensión de cualquier situación es la capacidad de predecir lo que tiene probabilidades de ocurrir a continuación. En la situación de la figura 44, prevemos que las personas que cruzan la calle seguirán caminando en la dirección que llevan y que la mujer seguirá agarrando el cochecito, la correa del perro y el teléfono. Podríamos predecir que la mujer tirará de la correa y el perro se resistirá para continuar su exploración de los aromas locales. La mujer tirará con más fuerza y el perro la seguirá, bajando de la acera a la calzada. Quien conduce tiene que estar preparado para ello. A un nivel aún más básico, prevemos que la mujer conservará los zapatos en los pies y la cabeza en el cuerpo, y que la calle seguirá en su sitio. Prevemos que el hombre saldrá de detrás del cochecito y tendrá piernas, pies y zapatos, que utilizará para subir a la acera. En definitiva, tenemos lo que los psicólogos llaman modelos mentales de aspectos importantes del mundo, basados en nuestro conocimiento de las realidades físicas y biológicas, la causa y el efecto y el comportamiento humano. Estos modelos —representaciones de cómo funciona el mundo—

nos permiten «simular» mentalmente situaciones. Los neurocientíficos no saben bien cómo surgen esos modelos mentales —o las simulaciones mentales que se «ejecutan» en ellos— a partir de la actividad de miles de millones de neuronas conectadas. Sin embargo, algunos psicólogos destacados han sugerido que la comprensión de conceptos y situaciones se produce precisamente a través de esas simulaciones mentales, es decir, al activar recuerdos de la propia experiencia física previa e imaginar qué acciones podemos emprender.[314]

Nuestros modelos mentales no solo nos permiten predecir lo que es probable que ocurra en una situación determinada, sino que también nos permiten imaginar lo que pasaría si se produjeran determinados hechos. Si el conductor toca el claxon o grita «¡Quítese de ahí!» por la ventanilla del coche, la mujer probablemente brincará sobresaltada y le prestará atención. Si tropieza y pierde el zapato, se agachará para recogerlo. Si el bebé del cochecito se pone a llorar, ella mirará a ver qué le pasa. Una parte fundamental de comprender una situación es ser capaz de utilizar los modelos mentales para imaginar distintos futuros posibles.[315]

La comprensión como simulación

El psicólogo Lawrence Barsalou es uno de los partidarios más conocidos de la hipótesis de «la comprensión como simulación». A su juicio, nuestra comprensión de las situaciones que encontramos consiste en hacer (inconscientemente) este tipo de simulaciones mentales. Además, Barsalou sugiere que estas simulaciones mentales también son la base de nuestra comprensión de situaciones en las que no participamos directamente, es decir, situaciones que podemos ver, oír o leer. Dice: «A medida que las personas comprenden un texto, construyen simulaciones que representan su contenido perceptivo, motor y afectivo. Las simulaciones parecen ser fundamentales para la representación del significado».[316]

Puedo imaginarme sin dificultad leyendo una noticia sobre, por ejemplo, un accidente de coche en el que se ve envuelta una mujer que cruza la calle mientras habla por teléfono, y puedo comprenderla mediante mi simulación mental de la situación. Quizá me pongo en el lugar de la mujer e imagino (mediante la simulación de mis modelos mentales) lo que se siente mientras sostiene el teléfono, empuja el cochecito, sujeta la correa del perro, cruza la calle, se distrae, etc.

¿Pero qué pasa con las ideas muy abstractas, como la verdad, la existencia o el infinito? Barsalou y sus colaboradores sostienen desde hace décadas que incluso los conceptos más abstractos los comprendemos mediante la simulación mental de situaciones concretas en las que se plasman esos conceptos. Según él, «el procesamiento conceptual utiliza reconstrucciones de estados sensoriomotores —simulaciones— que representan categorías»,[317] incluso las más abstractas. Sorprendentemente (al menos para mí), algunas de las pruebas que respaldan de forma más convincente esta hipótesis proceden del estudio cognitivo de la metáfora.

Metáforas de la vida cotidiana

Hace mucho tiempo, en una clase de Lengua, aprendí la definición de *metáfora*, que era más o menos así:

> Una metáfora es una figura retórica que describe un objeto o una acción de un modo que no es literal, pero que ayuda a explicar una idea o a establecer una comparación. […] Las metáforas se utilizan en la poesía, la literatura y siempre que alguien quiere adornar un poco su vocabulario.[318]

Mi profesor de Lengua nos dio ejemplos de metáforas, incluidos los versos más famosos de Shakespeare. «¿Qué luz alumbra aquella ventana? / Es el este, y Julieta es el sol». O «La vida es una sombra que camina, un mal actor / que en escena se inquieta y contonea / y nunca más se le oye». Y así sucesivamente. Me quedé con la idea de que la metáfora se utilizaba sobre todo para condimentar lo que, sin ella, podría ser anodino.[319]

Muchos años después, leí el libro *Metáforas de la vida cotidiana*,[320] escrito por el lingüista George Lakoff y el filósofo Mark Johnson. Mi anterior concepción de la metáfora sufrió un vuelco (si me perdonan la metáfora). La tesis de Lakoff y Johnson es que nuestro lenguaje cotidiano no solo está repleto de metáforas que suelen sernos invisibles, sino que nuestra comprensión de básicamente todos los conceptos abstractos se basa en metáforas derivadas de conocimientos físicos básicos. Lakoff y Johnson aportan pruebas de su tesis en forma de una amplia colección de ejemplos lingüísticos, que demuestran cómo conceptualizamos conceptos abstractos como *tiempo*, *amor*, *tristeza*, *ira* y *pobreza* usando términos de conceptos físicos concretos.

Por ejemplo, Lakoff y Johnson señalan que hablamos del concepto abstracto de *tiempo* con términos que se aplican al concepto más concreto de *dinero*. «Gastamos» o «ahorramos» tiempo. A menudo «no podemos desperdiciar el tiempo». A veces el tiempo que gastamos «vale la pena» y hemos «utilizado el tiempo de forma provechosa». Quizá conocemos a alguien que tiene «los días contados».

Del mismo modo, conceptualizamos estados emocionales como la felicidad y la tristeza en forma de direcciones físicas, hacia arriba y hacia abajo. Podemos «sentirnos hundidos» y «caer en una depresión». Nuestro estado de ánimo puede «caer a toda velocidad». Nuestros amigos suelen «levantarnos el ánimo» y nos dejan con «la moral alta».

Si vamos más allá, solemos conceptualizar las relaciones sociales en términos de temperatura física. «Me dieron una cálida bienvenida». «Me miró con frialdad». «Me trató fríamente». Estas expresiones están tan asentadas que no nos damos cuenta de que estamos hablando en lenguaje metafórico. La afirmación de Lakoff y Johnson de que estas metáforas revelan la base física de nuestra comprensión de los conceptos apoya la teoría de Lawrence Barsalou de que comprendemos mediante la simulación de modelos mentales construidos a partir de nuestro conocimiento básico.

Los psicólogos han investigado estas ideas a través de muchos experimentos fascinantes. Un grupo de científicos observó que la zona del cerebro que se activa cuando una persona piensa en el calor físico parece ser la misma que cuando piensa en el calor social. Para investigar las posibles consecuencias psicológicas, los investigadores llevaron a cabo un experimento con un grupo de sujetos voluntarios. Cada sujeto hizo un corto viaje en ascensor, acompañado por un miembro del equipo, hasta el laboratorio de psicología. Durante el trayecto, el miembro del laboratorio pedía al sujeto que sostuviera una taza de café caliente o helado «durante unos segundos» mientras él escribía el nombre de esa persona. Los sujetos no sabían que eso formaba parte del experimento. En el laboratorio, cada sujeto leía una breve descripción de una persona ficticia y se le pedía que valorara varios rasgos de su personalidad. Los que habían sostenido el café caliente en el ascensor consideraron a la persona de ficción mucho «más cálida» que los que habían sostenido el café helado.[321]

Otros investigadores han obtenido resultados similares. Además, esta vinculación entre «temperatura» física y social también parece existir a la inversa: otros psicólogos han descubierto que las experiencias sociales «cálidas» o «frías» hacen que los sujetos sientan más calor o frío físico.[322]

Aunque estos experimentos e interpretaciones siguen siendo objeto de controversia en el mundo de la psicología, se puede interpretar que los resultados respaldan las teorías de Barsalou y de Lakoff y Johnson: entendemos conceptos abstractos en términos de conocimientos físicos básicos. Si se activa mentalmente el concepto de *calidez* en sentido físico (por ejemplo, al sostener una taza de café caliente), se activa también el concepto de *calidez* en sentido más abstracto y metafórico, como al juzgar la personalidad de alguien, y viceversa.

Es difícil hablar de comprensión sin hablar de conciencia. Cuando empecé a escribir este libro, tenía pensado evitar por completo la cuestión de la conciencia, porque está llena de problemas científicos. Pero he

decidido que me voy a permitir especular un poco. Si nuestra comprensión de conceptos y situaciones consiste en realizar simulaciones utilizando modelos mentales, quizá el fenómeno de la conciencia —y toda nuestra concepción del yo— proviene de nuestra capacidad para construir y simular modelos de nuestros propios modelos mentales. No solo puedo simular mentalmente, por ejemplo, el acto de cruzar la calle mientras hablo por teléfono, sino que puedo simularme mentalmente a mí misma pensándolo y puedo predecir lo que quizá voy a pensar a continuación. Tengo un modelo de mi propio modelo. Modelos de modelos, simulaciones de simulaciones: ¿por qué no? Y así como la percepción física del calor, por ejemplo, activa una percepción metafórica del calor y viceversa, nuestros conceptos relacionados con las sensaciones físicas pueden activar el concepto abstracto del yo, que se retroalimenta a través del sistema nervioso para producir una percepción física del yo, o de la conciencia, si se prefiere. Esta causalidad circular es similar a lo que Douglas Hofstadter llamaba el «extraño bucle» de la conciencia, «en el que los niveles simbólico y físico se retroalimentan mutuamente y vuelven la causalidad del revés, de forma que parece que los símbolos tienen libre albedrío y han adquirido la capacidad paradójica de mover las partículas, en lugar de lo contrario».[323]

Abstracción y analogía

Hasta ahora he descrito varias ideas de la psicología sobre el conocimiento «intuitivo» básico con el que los seres humanos nacen o que adquieren en las primeras etapas de la vida, y cómo este conocimiento básico es el fundamento de los modelos mentales que forman nuestros conceptos. La construcción y el uso de esos modelos mentales se basan en dos capacidades humanas fundamentales: la abstracción y la analogía.

La abstracción es la capacidad de identificar conceptos y situaciones particulares como casos de una categoría más general. Concretemos más la idea de abstracción (perdón por el juego de palabras). Imaginemos a un

hombre que es padre y es psicólogo cognitivo. Digamos que su hija se llama S. El padre observa crecer a S. y escribe un diario sobre su capacidad de abstracción, cada vez más sofisticada. Imaginemos algunas de las anotaciones de su diario a lo largo de los años.

Tres meses: S. puede distinguir entre expresiones faciales que representan felicidad y tristeza y las generaliza entre las diversas personas con las que se relaciona. Ha abstraído los conceptos de cara feliz y cara triste.

Seis meses: S. ya reconoce cuándo la gente le está diciendo «adiós con la mano» y puede hacer lo mismo. Ha abstraído el concepto visual de saludar con la mano y ha aprendido a responder con el «mismo» gesto.

Dieciocho meses: S. ha abstraído los conceptos de *gato* y *perro* (y muchas otras categorías) de modo que es capaz de reconocer diferentes ejemplos de gatos y perros en fotografías, dibujos y caricaturas, así como en la vida real.

Tres años: S. reconoce letras sueltas del abecedario escritas por diferentes personas e impresas. Es más, distingue entre minúsculas y mayúsculas. Sus abstracciones de conceptos relacionados con las letras son muy avanzadas. Además, ha generalizado su conocimiento de las zanahorias, el brócoli, las espinacas, etc., al concepto más abstracto de *verdura*, que ahora equipara con otro concepto abstracto: *asqueroso*.

Ocho años: He oído por casualidad a J., la mejor amiga de S., contándole a S. que su madre se había olvidado de recogerla después del partido de fútbol. S. contestó: «Ah, sí, a mí me pasó exactamente lo mismo. Seguro que te enfadaste y tu madre lo pasó fatal». Me he dado cuenta de que ese «exactamente lo mismo» era en realidad una situación bastante diferente de la que se produjo cuando la cuidadora de S. se olvidó de recogerla en el colegio para llevarla a una clase de piano. Al decir «me pasó exactamente lo mismo», es evidente que S. ha construido un concepto abstracto que consiste más o menos en que una persona que cuida a una niña se olvida de recogerla antes o después de una actividad. Además, S. es capaz de seguir

una línea a partir de su propia experiencia para predecir cómo habrán reaccionado J. y la madre de J.

Trece años: S. empieza a ser una adolescente rebelde. Le he pedido muchas veces que limpie su habitación. Hoy me ha gritado: «No puedes obligarme; ¡Abraham Lincoln liberó a los esclavos!». Me he enfadado, sobre todo por su mala analogía.

Dieciséis años: El interés de S. por la música va en aumento. A los dos nos gusta jugar en el coche a poner una emisora de música clásica a mitad de una pieza y ver quién de los dos puede averiguar antes el compositor o la época de la pieza. Todavía se me da mejor a mí, pero S. está aprendiendo muy bien a reconocer el concepto abstracto de *estilo musical*.

Veinte años: S. me ha enviado un largo mensaje de correo electrónico sobre su vida en la universidad. Describe su semana como «un *estudiomaratón*, seguido de una *comidamaratón* y un *sueñomaratón*». Dice que la universidad la está convirtiendo en una «cafeadicta». En el mismo correo menciona una protesta estudiantil por el supuesto encubrimiento por parte de la universidad del presunto comportamiento sexual inapropiado de un profesor muy prestigioso; dice que los estudiantes llaman a la situación «acoso-*gate*». Seguramente, S. ni siquiera se da cuenta de que su mensaje ofrece grandes ejemplos de una forma común de abstracción en el lenguaje: formar nuevas palabras uniendo varias palabras o añadiendo sufijos para denotar situaciones abstractas. Añadir *maratón* designa una actividad de duración o cantidad excesivas; añadir *adicta* significa que está enganchada; y añadir -*gate* (de Watergate) significa un escándalo o encubrimiento.[324]

Veintiséis años: S. se ha licenciado en Derecho y la han contratado en un prestigioso bufete. Su cliente más reciente (la parte demandada) es una empresa de internet que ofrece una plataforma pública de blogs. La empresa recibió una demanda por difamación de un hombre (el demandante), porque un bloguero que utiliza la plataforma de la empresa escribió comentarios difamatorios sobre él. S. alegó ante el jurado que la plataforma de blogs es

como un «muro» en el que «varias personas han decidido hacer pintadas» y que la empresa no es más que la «propietaria del muro», por lo que no es responsable. Su argumento convenció al jurado, que falló en favor de la parte demandada. Es su primera gran victoria en los tribunales.[325]

El propósito de mi incursión en el diario imaginario de un padre es exponer algunas cuestiones importantes sobre la abstracción y la analogía. La abstracción, en cierto sentido, es la base de todos nuestros conceptos, incluso desde la más tierna infancia. Algo tan elemental como reconocer la cara de la madre —en diferentes condiciones de luz, diferentes ángulos, con diferentes expresiones faciales o diferentes peinados— es una proeza de abstracción en la misma medida que reconocer un estilo musical o hacer una analogía jurídica convincente. Como ilustran estas anotaciones, todo lo que llamamos percepción, categorización, reconocimiento, generalización y recuerdo («a mí me pasó exactamente lo mismo») entraña el acto de abstraer las situaciones que experimentamos.

La abstracción está estrechamente unida a la construcción de analogías. Douglas Hofstadter, que ha estudiado la abstracción y la construcción de analogías desde hace décadas, define esta última de forma muy general como «la percepción de una esencia común entre dos cosas».[326] Esta esencia común puede ser un concepto que tiene nombre (por ejemplo, «cara feliz», «saludar con la mano», «gato» o «música barroca»), en cuyo caso lo llamamos categoría, o un concepto difícil de verbalizar y creado sobre la marcha (por ejemplo, un cuidador que se olvida de recoger a una niña antes o después de una actividad, o un propietario de un «espacio de escritura» público que no es responsable de lo que allí se «escribe»), en cuyo caso lo llamamos analogía. Estos fenómenos mentales son las dos caras de una misma moneda. En algunos casos, una idea como «las dos caras de la misma moneda» empezará siendo una analogía pero acabará entrando en nuestro vocabulario como modismo, lo que hace que la tratemos más bien como una categoría.

En resumen, las analogías, casi siempre inconscientes, son la base de nuestra capacidad de abstraer y formar conceptos. Como decían Hofstadter y su coautor, el psicólogo Emmanuel Sander: «Sin conceptos no puede haber pensamiento, y sin analogías no puede haber conceptos».[327]

En este capítulo he esbozado algunas ideas procedentes de trabajos recientes de psicología sobre los mecanismos mentales con los que los seres humanos comprenden y actúan de forma apropiada en las situaciones con las que se encuentran. Tenemos un conocimiento básico, en parte innato y en parte aprendido durante el desarrollo y durante toda la vida. Nuestros conceptos están codificados en el cerebro como modelos mentales que podemos «ejecutar» (es decir, simular) para predecir lo que es probable que ocurra en cualquier situación o lo que podría ocurrir si sucede cualquier alteración que imaginemos. Nuestros conceptos, que van desde simples palabras hasta situaciones complejas, se forman mediante abstracción y analogía.

Desde luego, no pretendo haber abarcado todos los elementos de la capacidad humana de comprensión. De hecho, muchas personas han señalado que los términos *comprensión* y *significado* (por no hablar de *conciencia*) no son más que términos mal definidos que utilizamos como referentes provisionales, porque todavía no disponemos del lenguaje ni de la teoría que necesitamos para hablar de lo que verdaderamente ocurre en el cerebro. Marvin Minsky, pionero de la inteligencia artificial, lo explicaba así: «Aunque los gérmenes de ideas precientíficas como *creer*, *saber* y *significar* son útiles en la vida diaria, técnicamente parecen demasiado toscos para sustentar teorías sólidas. […] Por muy reales que nos parezcan hoy *yo* o *comprender*, […] no son más que los primeros pasos hacia conceptos mejores». Minsky proseguía señalando que nuestras confusiones sobre estas nociones «nacen de una carga de ideas tradicionales insuficientes para este empeño tan tremendamente difícil. […] Estamos todavía en un periodo formativo de nuestras ideas sobre la mente».[328]

Hasta hace poco, la cuestión de los mecanismos mentales que permiten a las personas comprender el mundo —y de si las máquinas también podrían tener esa capacidad de comprensión— era competencia casi exclusiva de filósofos, psicólogos, neurocientíficos e investigadores teóricos de la IA, que han mantenido debates académicos sobre estas cuestiones durante décadas (y en algunos casos siglos), sin prestar demasiada atención a las consecuencias en el mundo real. Sin embargo, como he descrito en capítulos anteriores, ahora se están utilizando ampliamente para aplicaciones del mundo real sistemas de IA que carecen de una comprensión similar a la humana. De repente, cuestiones que antes quedaban circunscritas al ámbito académico han empezado a tener mucha importancia en el mundo real. ¿Hasta qué punto necesitan los sistemas de IA una comprensión similar a la humana, o algo que se aproxime, para hacer su trabajo de forma fiable y sólida? Nadie sabe la respuesta. Pero prácticamente todos los investigadores en IA están de acuerdo en que el conocimiento básico «de sentido común» y la capacidad de abstracción y analogía sofisticadas están entre los eslabones que faltan para que la IA siga avanzando. En el capítulo siguiente describo algunas estrategias para dotar a las máquinas de estas capacidades.

---

[310] G.-C. Rota, «In Memoriam of Stan Ulam: The Barrier of Meaning», *Physica D Nonlinear Phenomena* 22 (1986), pp. 1-3.

[311] En una conferencia que di sobre este tema, un estudiante preguntó: «¿Por qué es necesario que un sistema de IA tenga una comprensión similar a la humana? ¿Por qué no podemos aceptar una IA con otro tipo de comprensión?». Aparte de que no tengo ni idea de lo que significaría «otro tipo de comprensión», lo que quiero decir es que si los sistemas de IA van a interactuar con los humanos en el mundo, necesitan comprender las situaciones con que se encuentren básicamente de la misma manera que los humanos.

[312] El término *core knowledge*, «conocimiento básico», lo han utilizado sobre todo la psicóloga Elizabeth Spelke y sus colaboradores; por ejemplo, véase E. S. Spelke y K. D. Kinzler, «Core Knowledge», *Developmental Science* 10, n.º 1 (2007), pp. 89-96. Muchos otros científicos cognitivos han analizado ideas similares.

[313] Los psicólogos utilizan el término *intuitivo* porque este conocimiento básico está muy arraigado en nuestras mentes desde los primeros años; se convierte en una obviedad para nosotros y, en su mayor parte, permanece en el subconsciente. Numerosos psicólogos han demostrado que hay aspectos de las creencias intuitivas típicas del ser humano sobre física, probabilidad y otras áreas que en realidad son erróneos. Véanse, por ejemplo, A. Tversky y D. Kahneman, «Judgment Under Uncertainty: Heuristics and Biases», *Science* 185, n.º 4157 (1974), pp. 1124-1131; y B. Shanon, «Aristotelianism, Newtonianism, and the Physics of the Layman», *Perception* 5, n.º 2 (1976), pp. 241-243.

[314] Lawrence Barsalou ofrece un argumento detallado a favor de esas simulaciones mentales en L. W. Barsalou, «Perceptual Symbol Systems», *Behavioral and Brain Sciences* 22 (1999), pp. 577-660.

[315] Douglas Hofstadter señala que cuando nos encontramos con una situación (o cuando la recordamos, la imaginamos o leemos sobre ella), la representación de esa situación en nuestra mente incluye un «halo» de posibles variaciones de esa situación que él llama «una esfera contrafactual implícita», que incluye «las cosas que nunca han pasado pero que no podemos evitar ver de todos modos». D. R. Hofstadter, *Metamagical Themas*, Nueva York: Basic Books, 1985, p. 247.

[316] L. W. Barsalou, «Grounded Cognition», *Annual Review of Psychology* 59 (2008), pp. 617-645.

[317] L. W. Barsalou, «Situated Simulation in the Human Conceptual System», *Language and Cognitive Processes* 18, n.os 5-6 (2003), pp. 513-562.

[318] A. E. M. Underwood, «Metaphors», *Grammarly* (blog), consultado el 17 de diciembre de 2018, www.grammarly.com/blog/metaphor.

[319] La primera cita es de *Romeo y Julieta*, acto II, escena II, y la segunda de *Macbeth* acto V, escena V. Las traducciones son de la traductora.

[320] G. Lakoff y M. Johnson, *Metaphors We Live By*, Chicago: University of Chicago Press, 1980 [trad. cast.: *Metáforas de la vida cotidiana*, Madrid: Cátedra, 2017].

[321] L. E. Williams y J. A. Bargh, «Experiencing Physical Warmth Promotes Interpersonal Warmth», *Science* 322, n.º 5901 (2008), pp. 606-607.

[322] C. B. Zhong y G. J. Leonardelli, «Cold and Lonely: Does Social Exclusion Literally Feel Cold?», *Psychological Science* 19, n.º 9 (2008), pp. 838-842.

[323] D. R. Hofstadter, *I Am a Strange Loop*, Nueva York: Basic Books, 2007 [trad. cast.: *Yo soy un extraño bucle*, Barcelona: Tusquets, 2008]. La cita es de la solapa del libro. Mi descripción también se hace eco de las ideas propuestas por el filósofo Daniel Dennett en su libro *Consciousness Explained*, Nueva York: Little, Brown, 1991 [trad. cast.: *La conciencia explicada*, Barcelona: Paidós, 1995].

[324] Este tipo de «productividad lingüística» se analiza en D. Hofstadter y E. Sander, *Surfaces and Essences: Analogy as the Fuel and Fire of Thinking*, Nueva York: Basic Books, 2013, p. 129, y en A. M. Zwicky y G. K. Pullum, «Plain Morphology and Expressive Morphology», en *Annual Meeting of the Berkeley Linguistics Society* (1987), 13:330-340.

[325] He tomado prestado este argumento de un caso legal real. Véase «Blogs as Graffiti? Using Analogy and Metaphor in Case Law», IdeaBlawg, 17 de marzo de 2012, www.ideablawg.ca/blog/2012/3/17/blogs-as-graffiti-using-analogy-and-metaphor-in-case-law.html.

[326] D. R. Hofstadter, «Analogy as the Core of Cognition», Presidential Lecture, Stanford University (2009), consultado el 18 de diciembre de 2018, www.youtube.com/watch?v=n8m7lFQ3njk.

[327] Hofstadter y Sander, *Surfaces and Essences*, p. 3.

[328] M. Minsky, «Decentralized Minds», *Behavioral and Brain Sciences* 3, n.º 3 (1980), pp. 439-440.

15

Conocimiento, abstracción y analogía en la inteligencia artificial

Desde los años cincuenta, muchas personas del mundo de la IA han investigado formas de incorporar aspectos cruciales del pensamiento humano —como el conocimiento intuitivo básico, la abstracción y la creación de analogías— a la inteligencia automática para conseguir que los sistemas de inteligencia artificial comprendan realmente las situaciones con las que se encuentran. En este capítulo voy a describir varios intentos en este sentido, incluidos algunos de mis propios trabajos pasados y actuales.

Conocimiento básico para ordenadores

En los primeros tiempos de la IA, antes de que el aprendizaje automático y las redes neuronales se impusieran, los investigadores de la IA codificaban manualmente las reglas y los conocimientos que iba a necesitar un programa para ejecutar sus tareas. A muchos de los primeros pioneros de la IA les parecía totalmente razonable que este método de «integración» pudiera absorber una parte del conocimiento de sentido común de los humanos suficiente para lograr una inteligencia de nivel humano en los ordenadores.

El intento más famoso y duradero de codificar manualmente el conocimiento de sentido común para incorporarlo a las máquinas es el proyecto Cyc de Douglas Lenat. Lenat, estudiante de doctorado y más tarde profesor del Laboratorio de IA de la Universidad de Stanford, se hizo famoso en la comunidad de investigadores de IA de los años setenta creando programas que simulaban la forma de inventar nuevos conceptos de los seres humanos, sobre todo en matemáticas.[329] Sin embargo, tras más de una década de trabajo sobre este tema, Lenat llegó a la conclusión de que para que la IA progresara verdaderamente, las máquinas debían tener sentido común. De manera que decidió crear una inmensa colección de datos sobre la realidad y las reglas lógicas que sirvieran a los programas para manejar esos datos y deducir los que necesitaban. En 1984, Lenat abandonó su puesto académico para fundar una empresa (hoy llamada Cycorp) dedicada a este objetivo.

El nombre *Cyc* (pronunciado 'saic') pretende evocar la palabra *enciclopedia*, pero, a diferencia de las enciclopedias con las que estamos familiarizados, el objetivo de Lenat era que Cyc contuviera todo el conocimiento no escrito que tienen los seres humanos o, por lo menos, el suficiente para que los sistemas de IA pudieran tener un nivel humano en visión, lenguaje, planificación, razonamiento y otras tareas.

Cyc es un sistema simbólico de IA como los que describí en el capítulo 1: una colección de afirmaciones («aserciones») sobre entidades particulares o conceptos generales, escritas en un lenguaje informático basado en la lógica. He aquí algunos ejemplos de aserciones de Cyc (traducidas del lenguaje lógico al normal):[330]

→ Una entidad no puede estar en más de un lugar al mismo tiempo.
→ Los objetos envejecen un año cada año.
→ Cada persona tiene una madre que es una persona de sexo femenino.

El proyecto Cyc también incluye complejos algoritmos para hacer inferencias lógicas sobre aserciones. Por ejemplo, Cyc puede determinar

que si estoy en Portland, no estoy en Nueva York, porque soy una entidad, Portland y Nueva York son lugares, y una entidad no puede estar en más de un lugar a la vez. Cyc también cuenta en su colección con muchos métodos para manejar aserciones incoherentes o inciertas.

Las aserciones de Cyc las han codificado manualmente unos humanos (en concreto, los empleados de Cycorp) o las ha deducido lógicamente el sistema a partir de aserciones existentes.[331] ¿Cuántas aserciones se necesitan para capturar el conocimiento de sentido común de los humanos? En una conferencia pronunciada en 2015, Lenat dijo que Cyc tenía en ese momento alrededor de quince millones de aserciones, y calculó que «probablemente tenemos en torno al 5 por ciento de lo que a la hora de la verdad vamos a necesitar».[332]

La filosofía en la que se fundamenta Cyc tiene mucho en común con la de los sistemas expertos de los primeros tiempos de la IA. Quizá recuerden lo que dije en el capítulo 2 sobre el sistema experto en diagnóstico médico MYCIN. Los desarrolladores de MYCIN entrevistaron a «expertos» —médicos— para obtener unas reglas que permitieran al sistema hacer diagnósticos. Después, los desarrolladores tradujeron esas reglas a un lenguaje informático basado en la lógica para que el sistema pudiera hacer inferencias lógicas. En Cyc, los «expertos» son personas que traducen manualmente sus conocimientos sobre el mundo a enunciados lógicos. La «base de conocimientos» de Cyc es mayor que la de MYCIN, y sus algoritmos de razonamiento lógico son más sofisticados, pero los proyectos se basan en la misma fe de fondo: que es posible capturar la inteligencia mediante reglas programadas por humanos que trabajen con una colección suficientemente amplia de conocimientos explícitos. En el panorama actual de la IA, dominado por el aprendizaje profundo, el proyecto Cyc es una de las últimas iniciativas de IA simbólica a gran escala que quedan.[333]

¿Es posible que, con el tiempo y el esfuerzo suficientes, los ingenieros de Cycorp consigan capturar todo el conocimiento humano, o al menos el

suficiente, independientemente de cuánto sea eso? Lo dudo. Si el conocimiento de sentido común es el conocimiento que todos los seres humanos tienen pero que no está escrito en ninguna parte, entonces gran parte de ese conocimiento es subconsciente; ni siquiera sabemos que lo tenemos. Ahí se incluye gran parte de nuestro conocimiento intuitivo básico sobre física, biología y psicología, en el que se apoyan todos nuestros conocimientos generales sobre el mundo. Si no somos conscientes de que sabemos algo, no podemos ser los «expertos» encargados de suministrar explícitamente ese conocimiento a un ordenador.

Además, como expliqué en el capítulo anterior, nuestro conocimiento de sentido común se rige por la abstracción y la analogía. Lo que llamamos sentido común no puede existir sin esas aptitudes. Pero la abstracción y la construcción de analogías no son habilidades humanas que puedan ser capturadas por el inmenso conjunto de datos de Cyc ni, en mi opinión, por inferencia lógica en general.

En el momento de escribir este libro, el proyecto Cyc se encuentra ya en su cuarta década. Tanto Cycorp como su empresa derivada, Lucid, están comercializando Cyc ofreciendo un menú de aplicaciones especializadas para empresas. Las webs de las dos empresas presentan «ejemplos de éxito»: aplicaciones de Cyc en finanzas, extracción de petróleo y gas, medicina y otras áreas concretas. En cierto sentido, la trayectoria de Cyc se parece a la de Watson de IBM: las dos empezaron como un intento de investigación fundamental en IA, de amplio alcance y grandes ambiciones, y terminaron como una serie de productos comerciales con mensajes publicitarios exagerados (por ejemplo, Cyc «aporta a los ordenadores una capacidad de comprensión y razonamiento de tipo humano»),[334] pero con un enfoque más limitado y poco general y con escasa transparencia sobre los verdaderos resultados y capacidades del sistema.

Hasta ahora, Cyc no ha tenido mucha repercusión en el mundo de la IA. Además, algunos profesionales han criticado seriamente ese enfoque. Por

ejemplo, el profesor de IA de la Universidad de Washington Pedro Domingos dijo que Cyc era «el fracaso más lamentable de la historia de la IA».[335] El especialista del MIT en robótica Rodney Brooks no fue mucho más amable: «Aunque [Cyc] ha sido una labor heroica, no ha conseguido un sistema de IA capaz de tener ni siquiera una sencilla comprensión del mundo».[336]

¿Y si proporcionáramos a los ordenadores el conocimiento subconsciente sobre el mundo que adquirimos en la infancia y la niñez, y que está en la base de todos nuestros conceptos? ¿Cómo podríamos, por ejemplo, enseñarle a un ordenador la física intuitiva de los objetos? Varios grupos de investigación se han propuesto hacerlo y están construyendo sistemas de IA capaces de aprender un poco sobre la física de causa-efecto del mundo, a partir de vídeos, videojuegos y otros tipos de realidad virtual.[337] Estas estrategias son interesantes, pero no han hecho más que dar pasos de bebé —en comparación con lo que sabe un bebé de verdad— hacia el desarrollo de un conocimiento básico intuitivo.

Cuando el aprendizaje profundo empezó a exhibir su extraordinaria serie de éxitos, muchas personas, dentro y fuera del mundo de la IA, pensaron con optimismo que estábamos cerca de conseguir una IA general de nivel humano. Sin embargo, como he descrito a lo largo de este libro, a medida que los sistemas de aprendizaje profundo se utilizan más, están mostrando fallos en su «inteligencia». Ni siquiera los sistemas más eficaces son capaces de generalizar fuera de su estrecho ámbito de especialización, formar abstracciones o aprender sobre las relaciones causa-efecto.[338] Además, sus errores no humanos y su vulnerabilidad ante los llamados ejemplos antagónicos demuestran que en realidad no entienden los conceptos que intentamos enseñarles. Se sigue debatiendo si estos fallos se pueden subsanar con más datos o redes más profundas, o si falta algo más fundamental.[339]

En los últimos tiempos he visto una especie de giro en la conversación: cada vez más, la comunidad de la IA está volviendo a hablar de la crucial importancia de proporcionar a las máquinas sentido común. En 2018, Paul Allen, cofundador de Microsoft, duplicó el presupuesto de su instituto de investigación, el Allen Institute for AI, específicamente para estudiar el sentido común. Los organismos oficiales de financiación también están empezando a cambiar: en 2018, la Agencia de Proyectos de Investigación Avanzada para la Defensa (Defense Advanced Research Projects Agency, DARPA), uno de los principales fondos de financiación del Gobierno estadounidense para la investigación en IA, hizo público su plan para proporcionar una financiación sustancial a la investigación sobre el sentido común en la IA, con esta declaración: «[Hoy en día] el razonamiento de las máquinas es estrecho y muy especializado; siguen sin tener capacidad de razonamiento general y de sentido común. El programa [de financiación] construirá representaciones del conocimiento más parecidas a las humanas —por ejemplo, representaciones basadas en la percepción—, para suministrar a las máquinas un razonamiento de sentido común sobre el mundo físico y los fenómenos espaciotemporales».[340]

La abstracción idealizada

«Formar abstracciones» era una de las habilidades fundamentales de la IA que figuraban en la propuesta de IA presentada en Dartmouth en 1955 y que describí en el capítulo 1. Sin embargo, hacer que las máquinas puedan formar abstracciones conceptuales similares a las humanas sigue siendo un problema casi completamente sin resolver.

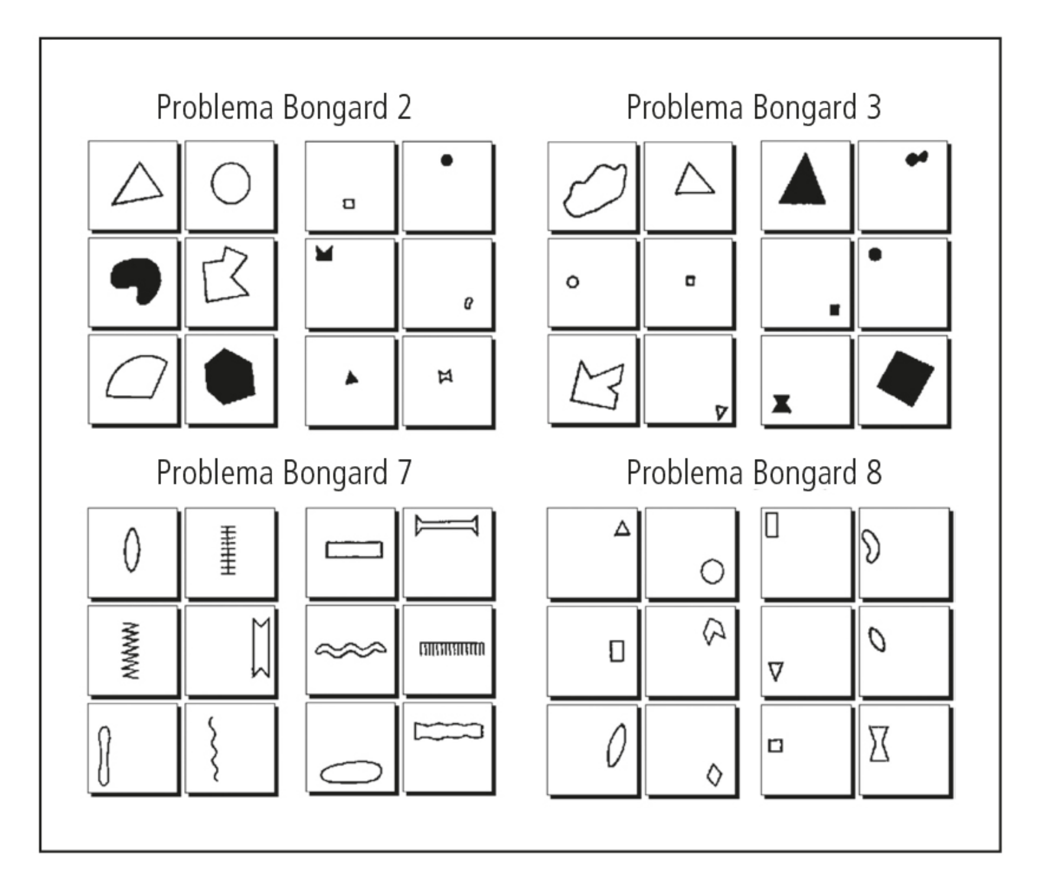


Figura 45. Cuatro ejemplos de problemas de Bongard. Para cada problema, la tarea consiste en determinar qué conceptos distinguen las seis casillas de la izquierda de las seis casillas de la derecha. Por ejemplo, para el problema 2 de Bongard, los conceptos son *grande* y *pequeño*.

La abstracción y la analogía son justo los temas que me llevaron inicialmente al campo de la IA. Me sentí interesada sobre todo cuando me topé con una serie de rompecabezas visuales llamados problemas de Bongard. Estos rompecabezas fueron formulados por un informático ruso, Mijaíl Bongard, que en 1967 publicó un libro (en ruso) titulado *Reconocimiento de patrones*.[341] Aunque el libro, en realidad, describía la propuesta de Bongard de un sistema parecido al perceptrón para el reconocimiento visual, la parte que más influencia ha tenido es el apéndice, en el que Bongard proponía cien problemas para que los resolvieran los programas de IA. La figura 45 muestra cuatro problemas del conjunto de Bongard.[342]

Cada problema tiene doce casillas: seis a la izquierda y seis a la derecha. Las seis casillas de la izquierda de cada problema plasman el «mismo» concepto, las seis casillas de la derecha plasman un concepto relacionado, y los dos conceptos diferencian a la perfección los dos conjuntos. El problema consiste en encontrar los dos conceptos. Por ejemplo, en la figura 45, los conceptos son (en el orden de las agujas del reloj) grande frente a pequeño; blanco frente a negro (o vacío frente a relleno, si se prefiere); lado derecho frente a lado izquierdo; y vertical frente a horizontal.

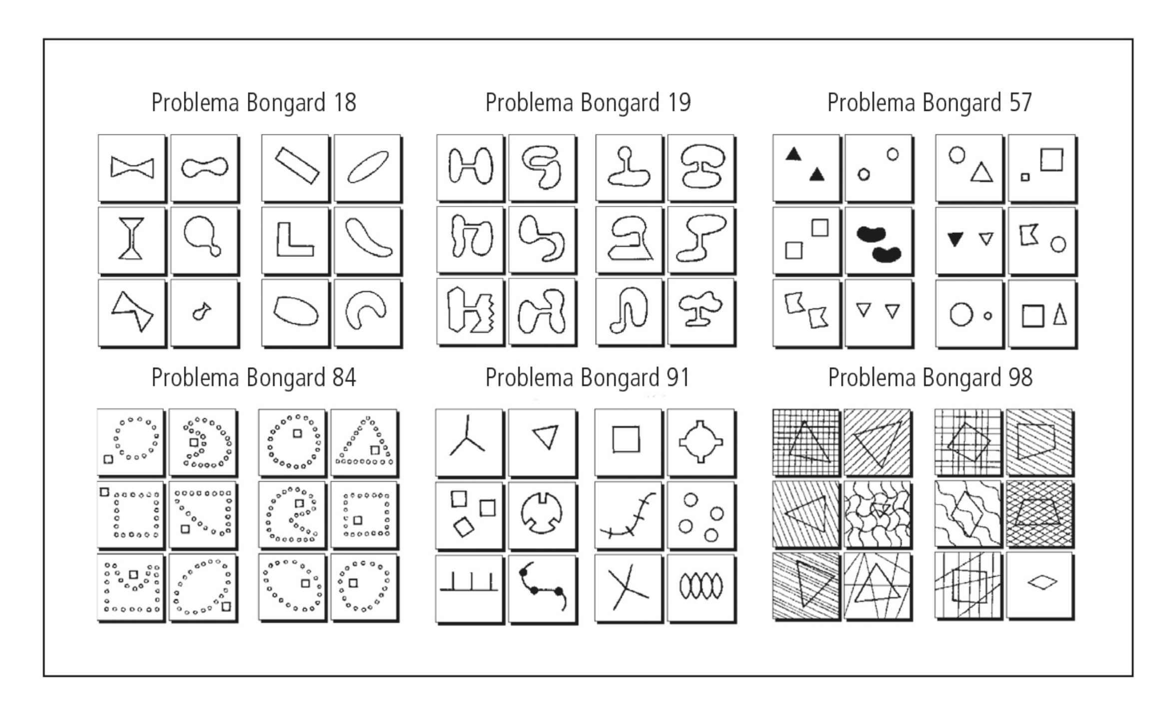


figura 46. Otros seis problemas de Bongard.

Los problemas de la figura 45 son relativamente fáciles de resolver. Bongard organizó sus cien problemas más o menos por orden de dificultad. Si quieren entretenerse, la figura 46 presenta otros seis problemas de la última parte de la lista. Daré las respuestas aquí a continuación.

Bongard diseñó cuidadosamente estos rompecabezas de forma que para resolverlos hicieran falta algunas de las capacidades de abstracción y construcción de analogías que un ser humano o un sistema de inteligencia artificial necesitan en el mundo real. En un problema de Bongard, se puede

considerar que cada una de las doce casillas es una «situación» idealizada en miniatura, con diferentes objetos, atributos y relaciones. Las situaciones de la izquierda comparten una «esencia» (por ejemplo, «grande»); las situaciones de la derecha comparten una esencia opuesta (por ejemplo, «pequeña»). Y en los problemas de Bongard, como en la vida real, identificar la esencia de una situación a veces es algo muy sutil. En palabras del científico cognitivo Robert French, la abstracción y la analogía consisten en percibir «la sutileza de la igualdad».[343]

Para descubrir esa sutil igualdad, hay que determinar qué atributos de la situación son relevantes y cuáles se pueden pasar por alto. En el problema 2 (figura 45), no importa si la figura es blanca o negra, ni dónde está colocada en la casilla, ni si es un triángulo, un círculo o ninguna otra cosa. Lo único que importa es el tamaño. Por supuesto, el tamaño no siempre importa; para los demás problemas de la figura 45, el tamaño es irrelevante. ¿Cómo sabemos discernir los humanos con tanta rapidez cuáles son los atributos relevantes? ¿Cómo podríamos conseguir que lo haga un ordenador?

Para complicar aún más las cosas a las máquinas, los conceptos relevantes pueden codificarse de forma abstracta y difícil de percibir, como pasa con los conceptos *tres* y *cuatro* del problema 91. En algunos problemas, puede que a un sistema de IA no le sea nada fácil averiguar qué cuenta como objeto, como en el problema 84 (*exterior* frente a *interior*), en el que los «objetos» relevantes están compuestos de objetos más pequeños (aquí, círculos pequeños). En el problema 98, los objetos están «camuflados»: los humanos pueden ver con facilidad qué son las figuras, pero las máquinas lo tienen más difícil, porque pueden tener dificultades para separar el primer plano del fondo.

Los problemas de Bongard también ponen a prueba nuestra capacidad de percibir nuevos conceptos sobre la marcha. El problema 18 es un buen ejemplo. El concepto común de las casillas de la izquierda no es fácil de verbalizar; es algo así como «objeto con una constricción o “cuello”». Pero

aunque nunca hayamos pensado en algo así, podemos reconocerlo rápidamente en el problema. Del mismo modo, en el problema 19, hay un nuevo concepto: algo así como «objeto con cuello horizontal» a la izquierda frente a «objeto con cuello vertical» a la derecha. Abstraer conceptos nuevos y difíciles de verbalizar —otro ejemplo de lo sutil que es la similitud— es algo que una persona hace muy bien, pero que ningún sistema de IA actual puede hacer de forma general.

El libro de Bongard, publicado en inglés en 1970, era bastante arduo y, al principio, poca gente supo de su existencia. Sin embargo, Douglas Hofstadter, que se había encontrado con el libro en 1975, quedó muy impresionado por los cien problemas del apéndice y escribió con detalle sobre ellos en su propio libro *Gödel, Escher, Bach*. Ahí es donde los vi por primera vez.

Desde niña, siempre me han gustado los rompecabezas, especialmente los que tienen que ver con la lógica o los patrones; cuando leí *GEB*, los problemas de Bongard me gustaron especialmente. También me intrigaron las ideas de Hofstadter esbozadas en *GEB* sobre cómo crear un programa que resolviera los problemas de Bongard emulando la percepción y la creación de analogías de los humanos. Es posible que la lectura de ese capítulo fuera lo que me empujó a ser investigadora de IA.

Los problemas de Bongard han cautivado a muchas otras personas, y varios investigadores han creado programas de IA que intentan resolverlos. La mayoría de estos programas hacen suposiciones simplificadoras (por ejemplo, limitan el conjunto de formas y relaciones de formas permitidas, o ignoran por completo los aspectos visuales y parten de una descripción de las imágenes elaborada por un humano). Cada programa ha conseguido resolver un subconjunto de problemas específicos, pero ninguno ha demostrado que sus métodos pudieran generalizarse como los de los humanos.[344]

¿Qué pasa con las redes neuronales convolucionales? Con los resultados tan espectaculares que han obtenido en la clasificación de objetos (por ejemplo, en el enorme desafío de reconocimiento visual de ImageNet que describí en el capítulo 5), ¿deberíamos confiar en que se pueda entrenar a una red de este tipo para resolver los problemas de Bongard? Sería imaginable plantear un problema de Bongard como una especie de problema de «clasificación» para una ConvNet, como se ilustra en la figura 47: las seis casillas de la izquierda podrían considerarse ejemplos de entrenamiento de la «clase 1», y las seis casillas de la derecha serían ejemplos de entrenamiento de la «clase 2». Ahora daríamos al sistema un nuevo ejemplo de «prueba». ¿Debe clasificarlo como clase 1 o clase 2?

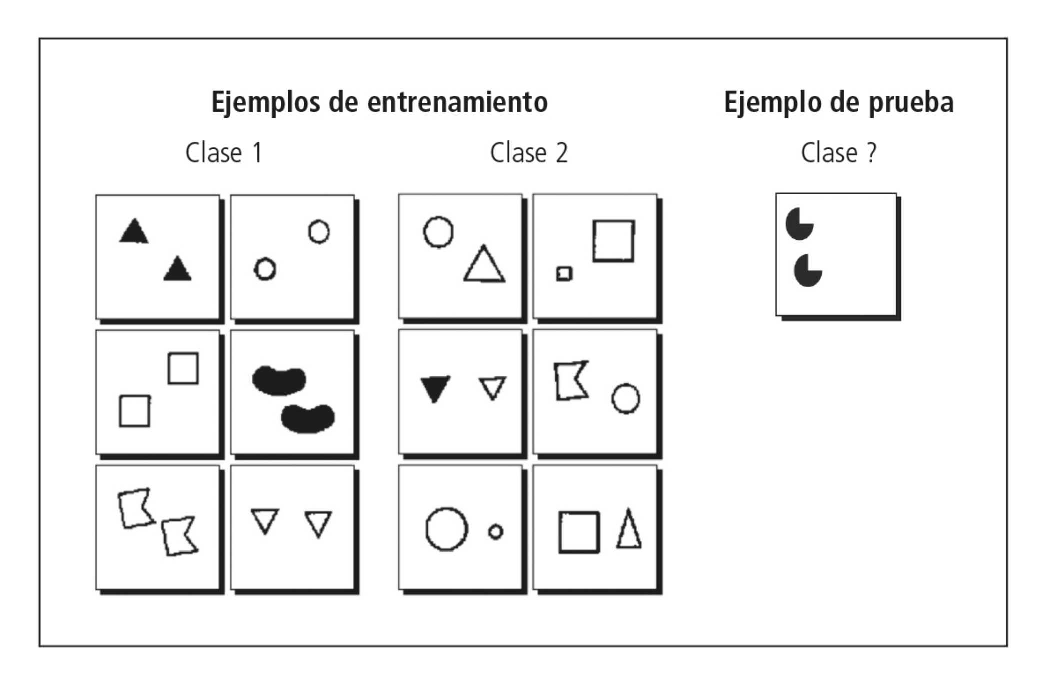


Figura 47. Una ilustración de cómo un problema de Bongard podría plantearse como un problema de clasificación, con doce ejemplos de entrenamiento y un nuevo ejemplo de «prueba».

Un obstáculo inmediato es que un conjunto de doce ejemplos de entrenamiento es ridículamente insuficiente para entrenar una ConvNet; ni siquiera mil doscientos bastarían. Por supuesto, eso es parte de lo que quiere plantear Bongard: los humanos podemos reconocer fácilmente los

conceptos relevantes con solo doce ejemplos. ¿Cuántos datos de entrenamiento necesitaría aprender una ConvNet para resolver un problema de Bongard? Aunque nadie ha hecho todavía un estudio sistemático sobre la resolución de problemas de Bongard con ConvNet, un grupo de investigadores analizó el rendimiento de las ConvNet de última generación en una tarea de diferenciar entre «igual frente a distinta», con imágenes similares a las de la figura 47.[345] La clase 1 incluía imágenes que tenían dos figuras de la misma forma; la clase 2 incluía imágenes con dos figuras de formas diferentes. Ahora bien, en lugar de usar doce imágenes, los investigadores entrenaron las ConvNet con veinte mil ejemplos para cada clase, la 1 («igual») y la 2 («distinta»). Tras el entrenamiento, se probó cada ConvNet con diez mil ejemplos nuevos. Todos los ejemplos se generaron automáticamente utilizando muchos tipos diferentes de formas. En los problemas de «igual frente a distinta», las ConvNet entrenadas no dieron más que unos resultados ligeramente superiores a las suposiciones aleatorias, mientras que los humanos evaluados por los autores obtuvieron una puntuación cercana al 100 por ciento. En resumen, las ConvNet actuales, aunque tienen una gran capacidad de aprender los elementos necesarios para reconocer objetos de ImageNet o elegir jugadas en go, no tienen las habilidades necesarias para hacer los tipos de abstracción y creación de analogías imprescindibles, ni siquiera en los problemas idealizados de Bongard, y mucho menos en el mundo real. Parece que los tipos de características que estas redes pueden aprender no bastan para formar esas abstracciones, por muchos ejemplos con los que se entrene una red. Las ConvNet no son las únicas que no tienen lo necesario: ningún sistema de IA actual tiene nada vagamente similar a estas capacidades humanas fundamentales.

Símbolos activos y construcción de analogías

Después de leer *Gödel, Escher, Bach* y decidir dedicarme a la investigación en IA, busqué a Douglas Hofstadter, con la esperanza de poder trabajar en algo parecido a los problemas de Bongard. Por suerte, después de insistir un poco, conseguí convencerle de que me permitiera entrar en su grupo de investigación. Hofstadter me explicó que su grupo estaba creando programas informáticos inspirados en la forma que tienen los seres humanos de comprender y establecer analogías entre situaciones. Tras licenciarse en Física (disciplina en la que la idealización, como el movimiento sin fricción, es un principio fundamental), Hofstadter estaba convencido de que la mejor forma de investigar un fenómeno —en este caso, cómo crean analogías los seres humanos— era estudiarlo en su forma más idealizada. La investigación en IA utiliza muchas veces los llamados micromundos: terrenos idealizados, como los problemas de Bongard, en los que un investigador puede desarrollar ideas antes de probarlas en ámbitos más complejos. Para su estudio sobre la construcción de analogías, Hofstadter desarrolló un micromundo aún más idealizado que los problemas de Bongard: rompecabezas de analogías con cadenas de letras. He aquí un ejemplo:

PROBLEMA 1: Supongamos que la cadena de letras *abc* se cambia por *abd*. ¿Cómo se cambiaría la cadena *pqrs* según «la misma regla»?

La mayoría de la gente responde «pqrt», porque deduce una regla que es más o menos «sustituir la letra más a la derecha por la que la sucede en el abecedario». Por supuesto, se podrían inferir otras reglas posibles, que generarían diferentes respuestas. Estas son algunas respuestas alternativas:

pqrd: «Sustituir la letra más a la derecha por *d*».
pqrs: «Sustituir todas las *c* por *d*. En *pqrs* no hay *c*, así que no cambia nada».
abd: «Sustituir cualquier cadena por la cadena *abd*».

Estas respuestas alternativas pueden parecer demasiado literales, pero no hay ningún argumento estrictamente lógico que diga que están mal. De

hecho, se podrían deducir infinitas reglas posibles. ¿Por qué la mayoría de la gente está de acuerdo en que una de ellas («pqrt») es la mejor? Parece que nuestros mecanismos mentales de abstracción —que evolucionaron para promover nuestra supervivencia y reproducción en el mundo real— se trasladan a este micromundo idealizado.

He aquí otro ejemplo:

> PROBLEMA 2: Supongamos que la cadena *abc* se cambia por *abd*. ¿Cómo se cambiaría la cadena *ppqqrrss* según «la misma regla»?

Incluso en este sencillo micromundo alfabético, la igualdad puede ser bastante sutil, al menos para una máquina. En el problema 2, una aplicación literal de la regla «sustituir la letra más a la derecha por la que la sucede» daría como resultado *ppqqrrst*, pero a la mayoría de la gente esta respuesta le parece demasiado literal; en lugar de ello, la gente tiende a responder «ppqqrrtt», porque percibe una correspondencia entre los pares de letras en *ppqqrrss* y las letras individuales en *abc*.[346] Los seres humanos tendemos bastante a agrupar objetos idénticos o similares.

El problema 2 ilustra, en este micromundo, la noción general de deslizamiento conceptual, una idea fundamental en la construcción de analogías.[347] Cuando se intenta percibir la «igualdad» esencial de dos situaciones diferentes, algunos conceptos de la primera situación tienen que «deslizarse», es decir, ser sustituidos por conceptos relacionados en la segunda situación. En el problema 2, el concepto *letra* se desliza a *grupo de letras*; por eso, la regla «sustituir la letra de más a la derecha por la que la sucede» se convierte en «sustituir el grupo de letras de más a la derecha por el que lo sucede».

Veamos ahora este problema:

> PROBLEMA 3: Supongamos que la cadena *abc* se cambia por *abd*. ¿Cómo se cambiaría la cadena *xyz* según «la misma regla»?

La mayoría de la gente responde «xya», porque se supone que la «sucesora» de *z* es *a*. Pero imaginemos un programa informático que no tiene el concepto de abecedario «circular» y para el que, por tanto, la letra *z* no tiene sucesora. ¿Qué otras respuestas serían razonables? Cuando se lo pregunté a la gente, obtuve muchas respuestas diferentes, algunas de ellas muy imaginativas. Curiosamente, las respuestas solían evocar metáforas físicas: por ejemplo, «xy» (la *z* «cae por el precipicio»), «xyy» (la *z* «rebota hacia atrás») y «wyz». La imagen que inspiraba esta última respuesta era que *a* y *z* están cada una «atrapadas contra una pared» en los extremos opuestos del abecedario, de modo que desempeñan papeles similares; por eso, si el concepto *primera letra del alfabeto* se desliza a *última letra del alfabeto*, entonces la letra más a la derecha se desliza a la letra más a la izquierda y la sucesora se desliza a la predecesora. El problema 3 ilustra cómo construir una analogía puede desencadenar una cascada de deslizamientos mentales.

El micromundo de las cadenas de letras hace muy visible la idea de deslizamiento. En otros ámbitos, puede ser más sutil. Por ejemplo, si nos fijamos en el problema 91 de Bongard de la figura 46, en el que la esencia común de las seis casillas de la izquierda es *tres*, los objetos que representan el concepto *tres* se deslizan de una casilla a otra; por ejemplo, de segmentos de línea (arriba a la izquierda) a cuadrados (en el centro a la izquierda), y luego a un concepto difícil de verbalizar en la casilla inferior izquierda (¿algo así como «las púas de un peine», quizá?). El deslizamiento conceptual también tenía un papel crucial en las diferentes abstracciones que la hija imaginaria S. (del capítulo anterior) hacía a lo largo de los años; por ejemplo, en su analogía jurídica, el concepto de *sitio web* se deslizaba al concepto de *muro*, y el concepto de *escribir un blog* se deslizó al concepto de *hacer pintadas con espray*.

Hofstadter concibió un programa informático, llamado Copycat, que resolvería este tipo de problemas utilizando algoritmos muy generales, similares a los que él creía que utilizaban los humanos cuando hacían

analogías en cualquier ámbito. El nombre *Copycat* procede de la idea de que el humano (el creador de analogías) debe resolver estos problemas «haciendo lo mismo», es decir, «copiando». La situación original (por ejemplo, *abc*) cambia de alguna manera, y entonces el sujeto tiene que hacer el «mismo» cambio en la nueva situación (por ejemplo, *ppqqrrss*).

Cuando me incorporé al grupo de investigación de Hofstadter, me encargó que trabajara con él en el desarrollo del programa Copycat. Como sabe cualquiera que haya pasado por ello, el camino hacia el doctorado consiste fundamentalmente en mucho trabajo salpicado de contratiempos frustrantes y (al menos en mi caso) una corriente subterránea constante de dudas sobre una misma. Pero de vez en cuando hay momentos de éxitos estimulantes, como cuando el programa en el que llevas cinco años trabajando por fin funciona. Voy a saltarme aquí todas las dudas, los contratiempos y las incontables horas de trabajo para ir directamente al final, cuando presenté mi tesis en la que describía el programa Copycat, que era capaz de resolver varias familias de problemas de analogías con cadenas de letras de forma (en mi opinión) similar a la humana.

Copycat no era ni un programa simbólico basado en reglas ni una red neuronal, aunque incluía aspectos de la IA simbólica y de la subsimbólica. Copycat resolvía problemas de analogía mediante una interacción continua entre los procesos de percepción del programa (es decir, darse cuenta de los elementos de un problema concreto de analogía con cadena de letras) y sus conceptos previos (por ejemplo, *letra*, *grupo de letras*, *sucesor*, *predecesor*, *igual* y *opuesto*). Los conceptos del programa estaban estructurados para emular algo parecido a los modelos mentales que describí en el capítulo anterior. En concreto, se basaban en el concepto de Hofstadter de «símbolos activos» en la cognición humana.[348] La arquitectura de Copycat era complicada y no voy a describirla aquí (aunque sí ofrezco algunas referencias sobre ella en las notas).[349] A la hora de la verdad, aunque Copycat podía resolver muchos problemas de analogía con cadena de letras

(incluidos los ejemplos que he presentado antes, además de muchas variaciones), el programa se limitaba a arañar la superficie de su ámbito, muy abierto. Por ejemplo, aquí hay dos problemas que mi programa no pudo resolver:

PROBLEMA 4: Si *azbzczd* cambia a *abcd*, ¿a qué cambia *pxqxrxsxt*?

PROBLEMA 5: Si *abc* cambia a *abd*, ¿a qué cambia *ace*?

Para ambos problemas es necesario reconocer nuevos conceptos sobre la marcha, una habilidad de la que Copycat carecía. En el problema 4, las *z* y las *x* desempeñan el mismo papel, algo así como «las letras de más que hay que borrar para ver la secuencia alfabética», lo que genera la respuesta «pqrst». En el problema 5, la secuencia *ace* es similar a la secuencia *abc*, salvo que en lugar de una secuencia de «sucesores» es una secuencia de «dobles sucesores», lo que genera la respuesta «acg». Me habría costado poco dotar a Copycat de la capacidad de contar el número de letras que hay, por ejemplo, entre *a* y *c* y *c* y *e*, pero no quería integrar aptitudes muy específicas del ámbito de las cadenas de letras. El objetivo de Copycat era que sirviera de banco de pruebas para ideas generales sobre la analogía, no que fuera un «creador de analogías entre cadenas de letras» integral.

Metacognición en el mundo de las cadenas de letras

Un aspecto esencial de la inteligencia humana —del que no se habla mucho en la IA hoy en día— es su capacidad de percibir y reflexionar sobre el propio pensamiento. En psicología, esto se llama metacognición. ¿Alguna vez han intentado en vano resolver un problema y han acabado reconociendo que estaban repitiendo los mismos procesos de pensamiento infructuosos? A mí me pasa con frecuencia, pero, una vez que identifico el patrón, a veces consigo salir del atasco. Copycat, como todos los demás programas de IA de los que he hablado en este libro, no tenía ningún

mecanismo de autopercepción, y eso perjudicaba su rendimiento. A veces, el programa se quedaba atascado por intentar una y otra vez resolver un problema de forma equivocada, y nunca se daba cuenta de que ya había recorrido un camino similar sin éxito.



Figura 48. Cuatro ejemplos sencillos de «pasear a un perro».

James Marshall, por aquel entonces estudiante de posgrado en el grupo de investigación de Douglas Hofstadter, asumió el proyecto de conseguir que Copycat reflexionara sobre su propio «pensamiento». Creó un programa llamado Metacat, que no solo resolvía problemas de analogía en el ámbito de las cadenas de letras que dominaba Copycat, sino que también trataba de ver patrones en lo que hacía. Cuando el programa se ejecutaba, generaba un comentario sobre los conceptos que reconocía en su propio proceso de resolución de problemas.[350] Como ocurría con Copycat,

Metacat tenía un comportamiento fascinante, pero se quedaba muy lejos de la capacidad de los humanos para reflexionar sobre sí mismos.

Identificar situaciones visuales

Mis investigaciones actuales se centran en el desarrollo de un sistema de IA que utilice la analogía para identificar con flexibilidad situaciones visuales, es decir, conceptos visuales que incluyan diversas entidades y sus relaciones. Por ejemplo, cada una de las cuatro imágenes de la figura 48 es un ejemplo de una situación visual que podríamos llamar «pasear a un perro». Es fácil de ver para los humanos, pero resulta que para los sistemas de IA, reconocer ejemplos de situaciones visuales, por sencillas que sean, resulta muy difícil. Identificar situaciones completas es mucho más difícil que identificar objetos individuales.






Figura 49. Cuatro casos atípicos de «pasear a un perro».

Mis colaboradores y yo estamos desarrollando un programa llamado Situate, que combina la capacidad de reconocimiento de objetos de las redes neuronales profundas con la arquitectura de símbolos activos de

Copycat, para identificar ejemplos de situaciones particulares a base de hacer analogías. Nos gustaría que nuestro programa fuera capaz de reconocer no solo ejemplos sencillos, como los de la figura 48, sino también ejemplos poco ortodoxos que requieran deslices conceptuales. El prototipo de situación «pasear a un perro» incluye una persona (que pasea al perro), un perro y una correa. La persona sujeta la correa, la correa está sujeta al perro y la persona y el perro pasean. ¿Verdad? En efecto, eso es lo que vemos en los ejemplos de la figura 48. Pero los humanos que entienden el concepto de pasear a un perro también pueden identificar cada una de las imágenes de la figura 49 como ejemplos de este concepto y, por otra parte, ser conscientes de cuánto «varía» cada una de ellas con respecto a la versión prototípica. El objetivo de Situate, que aún está en las primeras fases de desarrollo, es poner a prueba las ideas sobre los mecanismos generales que sirven de base a los humanos para crear analogías y demostrar que las ideas fundamentales del programa Copycat pueden funcionar con éxito más allá del micromundo de las analogías en cadenas de letras.

Copycat, Metacat y Situate no son más que tres ejemplos de los diversos programas de construcción de analogías basados en la arquitectura de símbolos activos de Hofstadter.[351] Es más, la arquitectura de símbolos activos es solo uno de los numerosos métodos que se están utilizando en el mundo de la IA para crear programas capaces de hacer analogías. Sin embargo, aunque la analogía es fundamental para la cognición humana a todos los niveles, todavía no existen programas de IA que se acerquen ni remotamente a la capacidad humana en este sentido.

«Estamos verdaderamente muy lejos»

La era moderna de la inteligencia artificial está dominada por el aprendizaje profundo, con su triunvirato de redes neuronales profundas, macrodatos y ordenadores ultrarrápidos. Sin embargo, en la búsqueda de una inteligencia

sólida y general, es posible que el aprendizaje profundo se esté encontrando con un muro: la importantísima «barrera del significado». En este capítulo he presentado un breve repaso de algunos de los intentos que se están haciendo en IA para derribar esa barrera. He analizado cómo los investigadores (yo entre ellos) están tratando de infundir en los ordenadores conocimiento de sentido común y capacidades de tipo humano para la abstracción y la creación de analogías.

Mientras reflexionaba sobre este tema, me llamó especialmente la atención una deliciosa y perspicaz entrada de blog escrita por Andrej Karpathy, el experto en aprendizaje profundo y visión por ordenador que ahora dirige los trabajos de IA en Tesla. En su artículo, titulado «The State of Computer Vision and AI: We Are Really, Really Far Away» (El estado de la visión por ordenador y la IA. Estamos verdaderamente muy lejos),[352] Karpathy describe sus reacciones, como investigador de visión por ordenador, ante una foto concreta, que se muestra en la figura 50. Dice que a los humanos esta imagen nos parece graciosa y se pregunta: «¿Qué haría falta para que un ordenador entendiera esta imagen como la entendemos usted o yo?».



Figura 50. La foto comentada en el blog de Andrej Karpathy.

Karpathy enumera muchas cosas que los humanos comprendemos fácilmente pero que siguen quedando fuera de alcance para incluso los mejores programas actuales de visión por ordenador. Por ejemplo, identificamos que hay gente en la escena, pero también que hay espejos, por lo que algunas de las personas son reflejos en esos espejos. Sabemos que el sitio es un vestuario y nos sorprende lo raro que es ver a un grupo de personas vestidas de traje en ese lugar.

Además, identificamos que una persona está de pie sobre una báscula, aunque esta consista en unos píxeles blancos que se confunden con el fondo. Karpathy señala que reconocemos que «Obama tiene el pie colocado ligeramente encima de la báscula» y que lo describimos sin problemas en términos de la estructura tridimensional de la escena que deducimos, en lugar de a partir de la imagen bidimensional que se nos da. Nuestro conocimiento intuitivo de la física nos permite razonar que el pie de Obama va a hacer que la báscula le atribuya más peso a la persona que está encima. Nuestro conocimiento intuitivo de la psicología nos dice que la persona que está en la báscula no es consciente de que Obama tiene también puesto el pie: lo deducimos por la dirección en la que mira y porque sabemos que no tiene ojos en la nuca. También sabemos que la persona probablemente no percibe la ligera presión del pie de Obama sobre la balanza. Nuestra teoría de la mente nos permite predecir además que el hombre no va a alegrarse cuando la báscula muestre que pesa más de lo que creía.

Por último, nos damos cuenta de que Obama y las demás personas que observan la escena están sonriendo: de sus expresiones deducimos que les divierte la broma que le está gastando Obama al hombre de la báscula, posiblemente más divertida debido a quién es Obama. También reconocemos que es una risa amistosa y que esperan que el hombre de la báscula también se ría cuando se dé cuenta. Karpathy señala: «Estás

razonando sobre el estado de ánimo de la gente y su opinión sobre el estado de ánimo de otra persona. La cosa se está volviendo aterradoramente meta».

En resumen: «Es asombroso que todas las inferencias anteriores se deriven de un breve vistazo a una matriz de valores [de píxeles] en dos dimensiones».

Creo que el ejemplo de Karpathy encarna a la perfección la complejidad de la comprensión humana, mostrando con claridad meridiana la magnitud de las dificultades que afronta la IA. Karpathy escribió el artículo en 2012, pero su mensaje sigue siendo válido hoy en día, y creo que lo seguirá siendo durante mucho tiempo.

Karpathy termina con esta reflexión:

> Una conclusión aparentemente ineludible para mí es que puede que […] necesitemos la corporalidad, y que la única manera de construir ordenadores que puedan interpretar escenas como nosotros es dejar que tengan todos los años de experiencia (estructurada, temporalmente coherente) que tenemos, la capacidad de interactuar con el mundo y una mágica arquitectura de aprendizaje activo e inferencia que casi no puedo ni imaginar cuando pienso en retrospectiva sobre lo que debería ser capaz de hacer.

En el siglo XVII, el filósofo René Descartes especuló con la idea de que nuestro cuerpo y nuestro pensamiento están formados por sustancias diferentes y sujetos a leyes físicas distintas.[353] Desde los años cincuenta, los enfoques dominantes para estudiar la IA han adoptado implícitamente la tesis de Descartes, asumiendo que unos ordenadores incorpóreos pueden llegar a alcanzar una inteligencia general. Sin embargo, un pequeño segmento de los profesionales de la IA ha defendido sistemáticamente la llamada hipótesis de la corporeidad: la premisa de que una máquina no puede alcanzar un nivel de inteligencia humano sin tener algún tipo de cuerpo que interactúe con el mundo.[354] Desde este punto de vista, un ordenador colocado sobre un escritorio o incluso un cerebro incorpóreo que creciera dentro de un contenedor no podría adquirir jamás los conceptos necesarios para la inteligencia general. Solo podría alcanzar una inteligencia de nivel humano una máquina adecuada, que tenga un cuerpo y

actúe en el mundo. Como Karpathy, casi no puedo ni imaginar qué avances harían falta para construir una máquina así. Pero, después de muchos años lidiando con la IA, el argumento de la corporalidad me parece cada vez más convincente.

---

[329] D. B. Lenat y J. S. Brown, «Why AM and EURISKO Appear to Work», *Artificial Intelligence* 23, n.º 3 (1984), pp. 269-294.

[330] Estos ejemplos son de C. Metz, «One Genius' Lonely Crusade to Teach a Computer Common Sense», *Wired*, 24 de marzo de 2016, www.wired.com/2016/03/doug-lenat-artificial-intelligence-common-sense-engine, y D. Lenat, «Computers Versus Common Sense», Google Talks Archive, consultado el 18 de diciembre de 2018, www.youtube.com/watch?v=gAtn-4fhuWA.

[331] Lenat señala que la empresa es cada vez más capaz de automatizar el proceso de obtención de nuevas aserciones (presumiblemente mediante la minería de la web). De D. Lenat, «50 Shades of Symbolic Representation and Reasoning», CMU Distinguished Lecture Series, consultado el 18 de diciembre de 2018, www.youtube.com/watch?v=4mv0nCS2mik.

[332] *Ibid.*

[333] Una descripción detallada y no técnica del proyecto Cyc figura en el capítulo 4 de H. R. Ekbia, *Artificial Dreams: The Quest for Non-biological Intelligence*, Cambridge, Reino Unido: Cambridge University Press, 2008.

[334] Página web de la empresa Lucid: lucid.ai.

[335] P. Domingos, *The Master Algorithm*, Nueva York: Basic Books, 2015, p. 35.

[336] De «The Myth of AI: A Conversation with Jaron Lanier», *Edge*, 14 de noviembre de 2014, www.edge.org/conversation/jaron_lanier-the-myth-of-ai.

[337] Por ejemplo, véanse N. Watters *et al.*, «Visual Interaction Networks», *Advances in Neural Information Processing Systems* 30 (2017), pp. 4539-4547; T. D. Ullman *et al.*, «Mind Games: Game Engines as an Architecture for Intuitive Physics», *Trends in Cognitive Sciences 21*, n.º 9 (2017), pp. 649-665; y K. Kansky *et al.*, «Schema Networks: Zero-Shot Transfer with a Generative Causal Model of Intuitive Physics», en *Proceedings of the International Conference on Machine Learning* (2017), pp. 1809-1818.

[338] J. Pearl, «Theoretical Impediments to Machine Learning with Seven Sparks from the Causal Revolution», en *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining* (2018), p. 3. Para profundizar en el razonamiento causal en la IA, véase J. Pearl y D. Mackenzie, *The Book of Why: The New Science of Cause and Effect*, Nueva York: Basic Books, 2018.

[339] Para un análisis inteligente de lo que falta en el aprendizaje profundo, véase G. Marcus, «Deep Learning: A Critical Appraisal», arXiv:1801.00631 (2018).

[340] DARPA, «Fiscal Year 2019 Budget Estimates», febrero de 2018, consultado el 18 de diciembre de 2018, www.darpa.mil/attachments/DARPAFY19PresidentsBudgetRequest.pdf.

[341] Versión en inglés: M. Bongard, *Pattern Recognition*, Nueva York: Spartan Books, 1970.

[342] Todas las imágenes de problemas de Bongard que presento aquí proceden del sitio web Index of Bongard Problems de Harry Foundalis, www.foundalis.com/res/bps/bpidx.htm, que recoge los cien problemas de Bongard, así como muchos problemas creados por otras personas.

[343] R. M. French, *The Subtlety of Sameness*, Cambridge, Mass.: MIT Press, 1995.

[344] Un programa especialmente interesante que intentaba resolver problemas de Bongard fue el que creó Harry Foundalis cuando era estudiante de posgrado en el grupo de investigación de Douglas Hofstadter en la Universidad de Indiana. Foundalis declaró que no estaba construyendo «un solucionador de problemas de Bongard», sino «una arquitectura cognitiva inspirada en los problemas de Bongard». El programa se inspiraba en la percepción humana a todos los niveles, desde la visión básica hasta la abstracción y la analogía, muy en consonancia con las intenciones de Bongard, aunque solo consiguió resolver unos cuantos problemas. Véase H. E. Foundalis, «Phaeaco: A Cognitive Architecture Inspired by Bongard's Problems», tesis doctoral, Universidad de Indiana, 2006, www.foundalis.com/res/Foundalis_dissertation.pdf. Foundalis mantiene una minuciosa web relacionada con su trabajo sobre los problemas de Bongard: www.foundalis.com/res/diss_research.html.

[345] S. Stabinger, A. Rodríguez-Sánchez y J. Piater, «25 Years of CNNs: Can We Compare to Human Abstraction Capabilities?», en *Proceedings of the International Conference on Artificial Neural Networks* (2016), pp. 380-387. Un estudio relacionado con resultados similares es el de J. Kim, M. Ricci y T. Serre, «Not-So-CLEVR: Visual Relations Strain Feedforward Neural Networks», *Interface Focus* 8, n.º 4 (2018): 2018.0011.

[346] Cuando digo «la mayoría de la gente» me refiero a los resultados de las encuestas que hice como parte de mi tesis doctoral. Véase M. Mitchell, *Analogy-Making as Perception*, Cambridge, Mass.: MIT Press, 1993.

[347] Hofstadter acuñó el término *deslizamiento conceptual* en su análisis de los problemas de Bongard en el capítulo 19 de D. R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, Nueva York: Basic Books, 1979.

[348] *Ibid.*, pp. 349-351.

[349] Hay una descripción detallada de Copycat en el capítulo 5 de D. R. Hofstadter y el Fluid Analogies Research Group, *Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought*, Nueva York: Basic Books, 1995. En el libro basado en mi tesis se ofrece una descripción todavía más detallada: Mitchell, *Analogy-Making as Perception*.

[350] J. Marshall, «A Self-Watching Model of Analogy-Making and Perception», *Journal of Experimental and Theoretical Artificial Intelligence* 18, n.º 3 (2006), pp. 267-307.

[351] Varios de estos programas se describen en Hofstadter y el Fluid Analogies Research Group, *Fluid Concepts and Creative Analogies*.

[352] A. Karpathy, «The State of Computer Vision and AI: We Are Really, Really Far Away», blog de Andrej Karpathy, 22 de octubre de 2012, karpathy.github.io/2012/10/22/state-of-computer-vision.

[353] Veáse *Stanford Encyclopedia of Philosophy*, s. v. «Dualism», plato.stanford.edu/entries/dualism/.

[354] Para un debate filosófico convincente sobre la hipótesis de la corporeidad en la ciencia cognitiva, véase A. Clark, *Being There: Putting Brain, Body, and World Together Again*, Cambridge, Mass.: MIT Press, 1996.

16

Preguntas, respuestas y especulaciones

Hacia el final de su libro de 1979 *Gödel, Escher, Bach*, Douglas Hofstadter se entrevistó a sí mismo sobre el futuro de la IA. En una sección titulada «Diez preguntas y especulaciones», planteaba y respondía preguntas no solo sobre las posibilidades del pensamiento automático, sino también sobre la naturaleza general de la inteligencia. Cuando leí *GEB*, recién licenciada, me interesó mucho esta parte. Las especulaciones de Hofstadter me convencieron de que, a pesar de todo el revuelo mediático sobre lo inminente de una inteligencia artificial de nivel humano (sí, ya lo hubo en los años ochenta), en realidad era un tema muy abierto y muy necesitado de nuevas ideas. A los jóvenes que empezábamos a trabajar en este campo aún nos aguardaban muchos obstáculos.

Al escribir ahora, más de tres décadas después, he pensado que sería apropiado acabar este libro con algunas de las preguntas, respuestas y especulaciones que yo me planteo, como homenaje a aquella sección de Hofstadter en *GEB* y como forma de enlazar las ideas que he presentado.

Pregunta: ¿Cuándo serán comunes y corrientes los coches autónomos?

Depende de lo que se entienda por *conducción autónoma*. La Administración Nacional de Seguridad del Tráfico en Carretera de Estados Unidos ha definido seis niveles de autonomía para los vehículos. Los reproduzco aquí parafraseados.

- → NIVEL 0: El conductor humano se encarga de toda la conducción.
- → NIVEL 1: El vehículo puede ayudar a veces al conductor humano con la dirección o la velocidad del vehículo, pero no con las dos simultáneamente.
- → NIVEL 2: El vehículo puede controlar simultáneamente la dirección y la velocidad del vehículo en algunas circunstancias (normalmente en autopista). El conductor humano debe seguir prestando atención («supervisar el entorno de conducción») en todo momento y encargarse de todos los demás aspectos de la conducción, como cambiar de carril, salir de las autopistas, detenerse en los semáforos y parar cuando lo ordena un coche de policía.
- → NIVEL 3: El vehículo puede encargarse de todos los aspectos de la conducción en determinadas circunstancias, pero el conductor humano debe prestar atención todo el tiempo y estar preparado para recuperar el control en cualquier momento en que el vehículo se lo pida.
- → NIVEL 4: El vehículo puede encargarse de todos los aspectos de la conducción en determinadas circunstancias. En esas circunstancias, el ser humano no necesita prestar atención.
- → NIVEL 5: El vehículo puede encargarse de toda la conducción en cualquier circunstancia. Los ocupantes humanos no son más que pasajeros y nunca necesitan encargarse de la conducción.[355]

Seguro que se han fijado en una condición muy importante: «en determinadas circunstancias». Es imposible hacer una lista exhaustiva de las circunstancias en las que, por ejemplo, un vehículo de nivel 4 puede encargarse de toda la conducción, mientras que es fácil imaginar muchas circunstancias que seguramente serían difíciles para un vehículo autónomo: por ejemplo, mal tiempo, congestión de tráfico urbano, una zona en obras o una carretera estrecha de doble dirección sin carriles señalizados.

En el momento de escribir estas líneas, la mayoría de los coches que circulan están entre los niveles 0 y 1: tienen control de velocidad de crucero, pero no de la dirección ni de los frenos. Algunos modelos recientes —los que tienen «control de crucero adaptativo»— se consideran de nivel 1. Hay ya algunos tipos de vehículos en los niveles 2 y 3, como los Tesla,

que disponen de un sistema de piloto automático. Los fabricantes y usuarios de estos vehículos están aprendiendo todavía qué situaciones se incluyen en las «determinadas circunstancias» en las que el conductor humano debe recuperar el control. También hay vehículos experimentales que pueden funcionar de forma totalmente autónoma en una gran variedad de circunstancias, pero siguen necesitando «conductores de seguridad» humanos que estén listos para tomar el control en un momento dado. Ha habido varios accidentes mortales causados por coches autónomos, entre ellos algunos experimentales, cuando se suponía que había una persona preparada para hacerse con el control pero en realidad no estaba prestando atención.

El sector de los coches autónomos quiere a toda costa fabricar y vender vehículos que lo sean por completo (es decir, de nivel 5); de hecho, la publicidad sobre los coches autónomos lleva mucho tiempo prometiéndonos la autonomía total a los consumidores. ¿Qué obstáculos impiden que nuestros coches tengan verdadera autonomía?

Los principales obstáculos son los tipos de situaciones de cola larga («casos extremos») que he descrito en el capítulo 6: situaciones para las que el vehículo no está entrenado y que, individualmente, quizá se den muy pocas veces, pero que, en conjunto, serán frecuentes cuando los vehículos autónomos se generalicen. Como ya he explicado, un conductor humano se enfrenta a estas situaciones utilizando el sentido común, sobre todo la capacidad de comprender y predecir situaciones nuevas por analogía con situaciones que el conductor ya conoce.

La plena autonomía de los vehículos también requiere el tipo de conocimiento intuitivo básico del que he hablado en el capítulo 14: intuición física, biológica y, sobre todo, psicológica. Para conducir con confianza en cualquier circunstancia, un conductor debe comprender las motivaciones, los objetivos e incluso las emociones de otros conductores, ciclistas, peatones y animales con los que comparte la carretera. Evaluar

una situación compleja y juzgar en milésimas de segundo quién parece que va a cruzar la calle en rojo, correr para coger el autobús, hacer un giro repentino sin señalizarlo o detenerse en pleno paso de peatones para ajustarse un zapato de tacón roto es algo que le sale de forma natural a la mayoría de los conductores humanos, pero todavía no a los coches autónomos.

Otro problema que amenaza a los vehículos autónomos es la posibilidad de que sufran ataques maliciosos de diversos tipos. Los expertos en seguridad informática han demostrado que ya muchos de los coches no autónomos que conducimos hoy en día —y que cada vez están más controlados por ordenador— son vulnerables a la piratería informática a través de sus conexiones a redes inalámbricas, como Bluetooth, redes de telefonía móvil y conexiones a internet.[356] Los coches autónomos, que estarán completamente controlados por ordenador, podrán quedar todavía más a merced de la piratería informática malintencionada. Además, como explicaba en el capítulo 6, los investigadores del aprendizaje automático han demostrado posibles «ataques antagónicos» a los sistemas de visión por ordenador de los coches autónomos, algunos tan sencillos como colocar discretas pegatinas en las señales de *stop* para que el coche las identifique como señales de límite de velocidad. Desarrollar medidas informáticas de seguridad adecuadas para los coches será una parte fundamental de la tecnología de conducción autónoma.

Aparte de los ataques informáticos, otro problema será lo que podríamos llamar la naturaleza humana. Será inevitable que la gente quiera gastar bromas a los coches autónomos para descubrir sus puntos débiles. Por ejemplo, subir y bajar de una acera (simulando que se va a cruzar la calle) para impedir que circule el coche. ¿Cómo habría que programar a los vehículos para que reconozcan este tipo de comportamientos y gestionen bien la situación? Además, con los vehículos totalmente autónomos

también habrá que resolver problemas legales de peso, como a quién se considera responsable en caso de accidente y qué tipo de seguro se exigirá.

Hay una cuestión especialmente delicada en relación con el futuro de los coches autónomos: ¿la industria debe aspirar a una autonomía parcial, en la que el coche conduzca en «determinadas circunstancias» pero el conductor humano tenga que prestar atención y recuperar el control en caso necesario? ¿O el único objetivo debe ser la autonomía total, que el ser humano pueda confiar por completo en la conducción del coche y no necesite prestar atención nunca?

La tecnología para fabricar vehículos totalmente autónomos y suficientemente fiables —que puedan manejarse por sí solos en casi todas las situaciones— no existe todavía debido a los problemas que he descrito antes. Es difícil prever cuándo se resolverán estos problemas; he visto que las predicciones de los «expertos» oscilan entre unos cuantos años y varias décadas. Hay que recordar la máxima de que el primer 90 por ciento de un proyecto tecnológico complejo ocupa el 10 por ciento del tiempo y el último 10 por ciento ocupa el 90 por ciento del tiempo.

La tecnología para la autonomía parcial de nivel 3 ya existe. Pero, como se ha demostrado muchas veces, los seres humanos no gestionan nada bien la autonomía parcial. Aunque los conductores humanos sepan que en teoría deben prestar atención en todo momento, a veces no lo hacen, y como los coches no son capaces de lidiar con todas las situaciones que se plantean, se producen accidentes.

¿En qué situación nos deja esto? Para lograr la plena autonomía en la conducción se necesita una IA general, que seguramente tardará todavía en ser realidad. Ya existen coches con autonomía parcial, pero son peligrosos porque los humanos que los conducen no siempre prestan atención. La solución más probable a este dilema es cambiar la definición de *autonomía total*: permitir que los coches autónomos conduzcan solo en zonas específicas, las que hayan sido creadas con la infraestructura necesaria para

garantizar que los coches van a ser seguros. Una versión frecuente de esta solución es la llamada «geovalla». Jackie DiMarco, exingeniera jefe de vehículos autónomos de Ford, lo explicaba así:

> Cuando hablamos de autonomía de nivel 4, es autonomía total dentro de una geovalla, es decir, dentro de una zona en la que tenemos delimitado un mapa de alta definición. Quien tiene ese mapa puede entender su entorno. Puede saber dónde están las farolas, dónde están los pasos de peatones, cuáles son las normas de circulación, el límite de velocidad, etc. Pensamos en una autonomía que se desarrolle dentro de una determinada geovalla y se vaya extendiendo a medida que avancen la tecnología, nuestro aprendizaje y nuestra capacidad para resolver cada vez más problemas.[357]

Claro que los inoportunos seres humanos siguen rondando por la geovalla. El investigador de IA Andrew Ng opina que hay que educar a los peatones para que tengan un comportamiento más predecible cerca de los vehículos autónomos: «Lo que le decimos a la gente es: “Por favor, respeten las normas y, por favor, sean considerados”».[358] La empresa de conducción autónoma de Ng, Drive.ai, ha creado una flota de furgonetas taxi totalmente autónomas que recogen y dejan pasajeros en zonas debidamente geovalladas; han empezado en Texas, uno de los pocos estados cuyas leyes permiten este tipo de vehículos. Pronto veremos los resultados del experimento, con sus optimistas planes de educación de los peatones.

Pregunta: ¿La IA supondrá el desempleo masivo entre los humanos?

No lo sé. Creo que no, al menos a corto plazo. La máxima de Marvin Minsky de que «lo fácil es difícil» sigue siendo cierta respecto a gran parte de la IA, y es probable que muchos trabajos humanos sean mucho más difíciles para los ordenadores (o los robots) de lo que se podría pensar.

No cabe duda de que los sistemas de IA sustituirán a los humanos en algunos trabajos; ya lo han hecho, y a menudo la sociedad ha salido beneficiada. Pero nadie sabe aún qué consecuencias globales tendrá la IA

en el empleo, porque nadie puede predecir las capacidades que tendrán las futuras tecnologías de IA.

Se han publicado muchas informaciones sobre los posibles efectos de la IA en el empleo que se centran sobre todo en la vulnerabilidad de los millones de puestos de trabajo relacionados con la conducción. Es posible que los humanos que trabajan en esos sectores acaben siendo sustituidos, pero la incertidumbre sobre cuándo se generalizará de verdad la conducción autónoma hace que sea difícil de predecir un calendario.

A pesar de la incertidumbre, la cuestión de la tecnología y el empleo forma parte (con razón) del debate general actual sobre la ética de la IA. Varias personas han señalado que, históricamente, las nuevas tecnologías han creado tantos tipos nuevos de puestos de trabajo como los que sustituyen, y quizá la IA no sea una excepción. Tal vez elimine puestos de conductor de camiones, pero la necesidad de desarrollar la ética de la IA hará que se creen nuevos puestos para filósofos morales. No digo esto para quitar importancia al posible problema, sino para dejar clara la incertidumbre que existe en torno a esta cuestión. Un minucioso informe elaborado en 2016 por el Consejo de Asesores Económicos de Estados Unidos sobre los posibles efectos de la IA en la economía subrayaba este aspecto: «Hay gran incertidumbre sobre cuánto y con qué rapidez se dejarán sentir estos efectos. […] Con las pruebas de que disponemos actualmente, no es posible hacer predicciones concretas, por lo que los responsables políticos deben estar preparados para una variedad de posibles resultados». [359]

Pregunta: ¿Podría ser creativo un ordenador?

A mucha gente la idea de que un ordenador sea creativo le parece un oxímoron. Al fin y al cabo, la esencia misma de una máquina es ser *mecánica*, un término que en el lenguaje cotidiano connota lo contrario de *creatividad*. Un escéptico podría razonar: «Un ordenador solo puede hacer

aquello para lo que está programado por un ser humano. Por tanto, no puede ser creativo; la creatividad implica crear algo nuevo por tu cuenta».[360]

Creo que este punto de vista —que un ordenador, por definición, no puede ser creativo porque solo puede hacer aquello para lo que está explícitamente programado— está equivocado. Hay muchas formas de que un programa de ordenador pueda generar cosas en las que nunca había pensado su programador. Mi programa Copycat (descrito en el capítulo anterior) creaba muchas veces analogías que nunca se me habrían ocurrido, pero que tenían su propia y peculiar lógica. Creo que, en principio, es posible que un ordenador sea creativo. Pero también creo que ser creativo implica ser capaz de comprender y juzgar lo que uno ha creado. Si tomamos la creatividad en este sentido, no se puede decir que ningún ordenador actual sea creativo.

Una pregunta relacionada es si un programa informático puede producir una obra de arte o una pieza musical bella. La belleza es muy subjetiva, pero mi respuesta es sí, sin la menor duda. He visto numerosas obras de arte generadas por ordenador que me parecen bellas. Un ejemplo es el «arte genético» del informático y artista Karl Sims.[361] Sims programaba ordenadores para que generaran obras de arte digitales utilizando un algoritmo inspirado en la selección natural darwiniana. A partir de funciones matemáticas con algunos elementos aleatorios, el programa de Sims generaba varias obras de arte posibles diferentes. Una persona seleccionaba la que más le gustase. El programa creaba variaciones de la obra seleccionada introduciendo elementos aleatorios en las funciones matemáticas de base. Después, la persona seleccionaba qué mutación prefería, y así sucesivamente durante muchas iteraciones. Este proceso creó algunas obras abstractas asombrosas, que se han expuesto en numerosos museos.

En el proyecto de Sims, la creatividad deriva del trabajo en equipo de humanos y ordenador: el ordenador genera primero unas obras de arte y

luego las sucesivas variaciones, y el ser humano juzga las obras creadas basándose en la comprensión humana de conceptos artísticos abstractos. El ordenador no comprende nada, así que por sí solo no es creativo.

Ha habido ejemplos similares con la creación de música, en los que un ordenador es capaz de generar música bella (o al menos agradable), pero en mi opinión la verdadera creatividad solo existe si colabora con un ser humano que preste la capacidad de entender qué hace que la música sea buena y, por tanto, emita un juicio sobre la producción del ordenador.

El programa informático más famoso que ha compuesto música de esta forma fue el programa Experiments in Musical Intelligence (Experimentos de Inteligencia Musical, EMI),[362] que mencioné en el prólogo. EMI se concibió para generar música según el estilo de varios compositores clásicos, y algunas de sus piezas consiguieron engañar incluso a músicos profesionales, a los que hizo creer que eran obra del verdadero compositor.

El programa EMI lo creó el compositor David Cope, en principio para que fuera una especie de «asistente de compositor» personal. A Cope le intrigaba la larga tradición de emplear la aleatoriedad para componer música. Un ejemplo famoso es el llamado juego de los dados musicales, practicado por Mozart y otros compositores del siglo XVIII, en el que un compositor cortaba una partitura musical en pequeños segmentos (por ejemplo, compases sueltos) y luego tiraba los dados para decidir cómo se ordenaban los segmentos en la nueva pieza.

Se podría decir que EMI era un juego de dados musicales elevado a la enésima potencia. Para que EMI creara piezas al estilo de Mozart, por ejemplo, Cope empezó por seleccionar de entre las obras de Mozart una gran colección de breves segmentos musicales y utilizó un programa informático escrito por él que identificaba patrones musicales cruciales que él llamaba «firmas»: los patrones que ayudan a definir el estilo peculiar del compositor. Cope escribió otro programa que clasificaba cada firma en función de las funciones musicales concretas que podía desempeñar en una

pieza. Las firmas se almacenaban en una base de datos correspondiente al compositor (Mozart, en nuestro ejemplo). Cope también desarrolló en EMI un conjunto de reglas —una especie de «gramática» musical— que plasmaban las restricciones a la hora de recombinar las variaciones de las firmas para crear una pieza musical coherente en un estilo concreto. EMI empleaba un generador de números aleatorios (el equivalente informático a lanzar dados) para seleccionar las firmas y crear segmentos musicales a partir de ellas; después, el programa utilizaba su gramática musical para decidir cómo ordenar los segmentos.

De este modo, EMI podía generar un número ilimitado de nuevas composiciones «al estilo» de Mozart o de cualquier otro compositor para el que se hubiera construido una base de datos de firmas musicales. Cope eligió cuidadosamente las mejores composiciones de EMI para publicarlas. He escuchado varias y me parece que las hay mediocres y sorprendentemente buenas, con algunos pasajes preciosos, aunque ninguna tiene la profundidad de la obra del compositor original. (Por supuesto, digo esto sabiendo de antemano que las piezas son de EMI, así que quizá tenga algún prejuicio). Las piezas más largas contienen muchos pasajes deliciosos, pero también tienen una tendencia poco humana a perder el hilo de una idea musical. No obstante, en general, las obras publicadas de EMI consiguen captar muy bien el estilo de varios compositores clásicos diferentes.

¿Era creativo EMI? Mi respuesta es no. Parte de la música generada por EMI era bastante buena, pero se basaba en los conocimientos musicológicos de Cope, que estaban integrados en las firmas musicales que él organizaba y las reglas musicológicas que él concebía. Y, sobre todo, yo diría que el programa no entendía realmente la música que generaba, ni en términos de conceptos musicales ni en cuanto a la repercusión emocional de la música. Por eso EMI no podía juzgar la calidad de su propia música. Esa

responsabilidad recaía en Cope, que se limitaba a decir: «Las obras que me gustan se publican y las que no, no».[363]

En 2005, en una decisión que me resulta desconcertante, Cope destruyó toda la base de datos de firmas musicales de EMI. El principal motivo que alegó fue que, como las composiciones de EMI eran tan fáciles e infinitas de generar, los críticos las despreciaban. Cope pensaba que solo valorarían a EMI como compositor si tenía, como escribió la filósofa Margaret Boden, «una obra finita, como la tienen todos los compositores humanos, asediados por la mortalidad».[364]

No sé si mi opinión le servirá de consuelo a Douglas Hofstadter, tan apenado por las composiciones más impresionantes de EMI y su capacidad de engañar a los músicos profesionales. Entiendo la preocupación de Hofstadter. Como ha observado el profesor de Literatura Jonathan Gottschall, «el arte es seguramente lo que más distingue a los seres humanos del resto de la creación. Es lo que más orgullosos de nosotros mismos nos hace sentir».[365] Pero me gustaría añadir que lo que nos enorgullece no es solo la creación de arte, sino también nuestra capacidad de apreciarlo, de entender por qué nos conmueve y de comprender lo que comunica. Esa apreciación y esa comprensión son esenciales tanto para el público como para el artista; sin ellas, creo que no se puede llamar «creativa» a una obra. En definitiva, a la pregunta «¿Podría ser creativo un ordenador?», respondería que en teoría sí, pero falta mucho tiempo.

Pregunta: ¿Cuánto tiempo falta para crear una IA general de nivel humano?

Para responder voy a citar a Oren Etzioni, director del Allen Institute for AI: «Las estimaciones que hayas hecho duplícalas, triplícalas, cuadruplícalas. Eso falta».[366]

Si quieren una segunda opinión, recordemos la frase de Andrej Karpathy del capítulo anterior: «Estamos verdaderamente muy lejos».[367]

Estoy de acuerdo.

Las computadoras empezaron siendo humanas. En realidad, solían ser mujeres que hacían cálculos a mano o con calculadoras mecánicas de escritorio; por ejemplo, los cálculos necesarios durante la Segunda Guerra Mundial para diseñar la trayectoria de los misiles y ayudar a los soldados a apuntar sus cañones de artillería. Ese era el significado original de *computadora*. Según el libro de Claire Evans *Broad Band*, en los años treinta y cuarenta, «el término *chica* y el término *computadora* eran intercambiables. Un miembro del Comité de Investigación de Defensa Nacional […] calculó que una unidad de "kilochica" de energía equivalía aproximadamente a mil horas de trabajo de computación».[368]

A mediados de los años cuarenta, las computadoras electrónicas sustituyeron a las humanas y enseguida se volvieron sobrehumanas: a diferencia de un ser humano, las máquinas podían calcular «la trayectoria de un proyectil muy veloz con más rapidez de la que tenía el propio proyectil».[369] Esta fue la primera de las muchas tareas concretas que las computadoras han hecho muy bien. Los ordenadores actuales —programados con algoritmos de IA de última generación— han conquistado muchas otras tareas concretas, pero la inteligencia general aún se les resiste.

Hemos visto que, durante toda la historia de este campo, diversos profesionales destacados han predicho que la IA general llegará de aquí a diez años, o quince, o veinticinco, o «dentro de una generación». Pero ninguna de estas predicciones se ha hecho realidad. Como expliqué en el capítulo 3, la «apuesta a largo plazo» entre Ray Kurzweil y Mitchell Kapor sobre si un programa superará o no una prueba de Turing cuidadosamente estructurada se decidirá en 2029. Yo apuesto por Kapor; estoy totalmente de acuerdo con su opinión, citada en el prólogo: «La inteligencia humana es un fenómeno maravilloso, sutil y mal conocido. No hay peligro de que se duplique a corto plazo».[370]

«Las predicciones son difíciles, especialmente sobre el futuro». Se puede discutir quién acuñó esta ingeniosa ocurrencia, pero vale para la IA tanto como para otros sectores. Varias encuestas entre especialistas en IA, a los que se preguntó cuándo llegaría la IA general o la IA «superinteligente», revelan una gran variedad de opiniones, que van desde «en los próximos diez años» hasta «nunca».[371] En otras palabras, no tenemos ni idea.

Lo que sí sabemos es que para que haya una IA general de nivel humano serán necesarias unas capacidades que los investigadores de la IA llevan décadas intentando comprender y reproducir —conocimiento de sentido común, abstracción y construcción de analogías, entre otras—, pero que han demostrado ser muy difíciles de emular. Y sigue habiendo otras preguntas sin respuesta: ¿la IA general necesitará tener consciencia?, ¿tener conciencia de sí misma?, ¿sentir emociones?, ¿poseer instinto de supervivencia y miedo a la muerte?, ¿tener un cuerpo? Como decía Marvin Minsky, al que cité anteriormente: «Estamos todavía en un periodo formativo de nuestras ideas sobre la mente».

La cuestión de cuándo alcanzarán los ordenadores la superinteligencia —«un intelecto mucho más capaz que los mejores cerebros humanos en prácticamente todos los campos, incluidas la creatividad científica, la sabiduría general y las habilidades sociales»—[372] me parece, como mínimo, problemática.

Varios autores han afirmado que si los ordenadores alcanzan una IA general de nivel humano, estas máquinas se volverán rápidamente «superinteligentes», en un proceso similar a lo que decía I. J. Good sobre una «explosión de inteligencia» (descrita en el capítulo 3). La teoría es que un ordenador con inteligencia general será capaz de leer todos los documentos de la humanidad y aprender todo lo que hay que saber a la velocidad del rayo. También será capaz de descubrir, gracias a unas capacidades de deducción cada vez mayores, todo tipo de nuevos conocimientos que podrá convertir en un nuevo poder cognitivo en sí

mismo. Esa máquina no estaría restringida por las molestas limitaciones que tenemos los humanos, como la lentitud de pensamiento y aprendizaje, la irracionalidad y los sesgos cognitivos, la susceptibilidad al aburrimiento, la necesidad de dormir y las emociones, que son estorbos para el pensamiento productivo. Desde este punto de vista, una máquina superinteligente abarcaría casi una inteligencia «pura», sin ninguna de nuestras debilidades.

Más probable me parece que estas supuestas limitaciones de los humanos sean una parte fundamental de nuestra inteligencia general. Es más, las limitaciones cognitivas que nos impone el hecho de tener un cuerpo que se mueve en el mundo, las emociones y los prejuicios «irracionales» que evolucionaron para permitirnos funcionar como grupo social y todas las demás cualidades consideradas a veces «deficiencias» cognitivas, son precisamente lo que nos permite ser inteligentes en sentido general y no sabios en lo concreto. No puedo demostrarlo, pero creo que es probable que la inteligencia general no pueda separarse de todas estas aparentes deficiencias, ni en los seres humanos ni en las máquinas.

En su sección «Diez preguntas y especulaciones» de *GEB*, Douglas Hofstadter abordó esta cuestión con una pregunta falsamente sencilla: «¿Podrá un ordenador pensante sumar deprisa?». Su respuesta me sorprendió la primera vez que la leí, pero ahora me parece acertada. «Quizá no. Nosotros también estamos hechos de un material que hace cálculos sofisticados, pero eso no significa que nuestro nivel simbólico, donde estamos "nosotros", sepa hacer esos mismos cálculos sofisticados. Por suerte para nosotros, nuestro nivel simbólico (es decir, nosotros) no puede acceder a las neuronas dedicadas a pensar, porque nos quedaríamos atontados. […] ¿Por qué no iba a pasar lo mismo con un programa inteligente?». Hofstadter seguía explicando que un programa inteligente, al igual que nosotros, representaría los números como «conceptos totalmente desarrollados, como hacemos nosotros, llenos de asociaciones […]. Con

todo este “lastre”, un programa inteligente sería bastante vago a la hora de sumar».[373]

Pregunta: ¿Hasta qué punto debemos estar aterrorizados por la IA?

Si lo que sabemos de la IA es lo que cuentan las películas y las novelas de ciencia ficción (e incluso algunos libros de divulgación), es normal que tengamos miedo de que la inteligencia cobre conciencia, se vuelva malvada e intente esclavizarnos o matarnos a todos. Pero, con lo lejos que parece estar todavía conseguir algo parecido a la inteligencia general, no es eso lo que preocupa a la mayoría de los profesionales de la IA. Como he expuesto en todo este libro, hay muchos motivos para preocuparse por la precipitación con la que nuestra sociedad está lanzándose hacia el uso de la tecnología de IA: la posibilidad de perder muchos puestos de trabajo, la posibilidad de que los sistemas de IA se usen de forma indebida y la inestabilidad y vulnerabilidad de estos sistemas frente a los ataques no son sino algunas de las preocupaciones muy legítimas de quienes se preocupan por las consecuencias de la tecnología en la vida de los seres humanos.

Comencé este libro relatando la consternación de Douglas Hofstadter ante los recientes avances de la IA, pero lo que le aterrorizaba a él, sobre todo, era otra cosa totalmente distinta. Lo que le preocupaba a Hofstadter era que a los programas de IA les resultara demasiado fácil igualar la cognición y la creatividad humanas, y que las sublimes creaciones de las mentes humanas que más veneraba —Chopin, por ejemplo— tuvieran que rivalizar con algoritmos superficiales como EMI, que utilizaban «todo tipo de trucos». Hofstadter se lamentaba: «Si esas mentes de infinita sutileza y complejidad y profundidad emocional pudieran acabar trivializadas por un pequeño chip, destruiría mi idea de cuál es la esencia de la humanidad». A Hofstadter también le inquietaban las predicciones de Kurzweil sobre la

futura Singularidad y le angustiaba que si Kurzweil tuviera razón, «nos desbancarán. Seremos reliquias. Nos quedaremos tirados».

Empatizo con Hofstadter y sus preocupaciones, pero creo que son claramente prematuras. El mensaje que más quiero transmitir con este libro es que los humanos tendemos a sobrevalorar los avances de la IA y a subestimar la complejidad de nuestra propia inteligencia. La IA actual está muy lejos de la inteligencia general, y no creo que la «superinteligencia» de las máquinas esté en el horizonte. Si la IA general se hace algún día realidad, estoy segura de que su complejidad será equiparable a la de nuestro cerebro.

En cualquier lista de preocupaciones a corto plazo sobre la IA, la superinteligencia debe estar muy abajo. En realidad, el auténtico problema es el contrario de la superinteligencia. A lo largo de este libro he mostrado que incluso los mejores sistemas de IA son frágiles; es decir, cometen errores cuando su entrada varía demasiado respecto a los ejemplos con los que se los ha entrenado. Muchas veces es difícil predecir en qué circunstancias saldrá a la luz la fragilidad de un sistema de IA. A la hora de transcribir un discurso, traducir entre idiomas, describir el contenido de fotos, conducir en una ciudad abarrotada, cuando es crucial un comportamiento fiable, sigue siendo necesaria la participación de los seres humanos. Creo que el aspecto más preocupante de los sistemas de IA a corto plazo es que les demos demasiada autonomía sin ser totalmente conscientes de sus limitaciones y vulnerabilidades. Tenemos tendencia a antropomorfizar la IA: le atribuimos cualidades humanas y acabamos sobrestimando hasta qué punto se puede confiar completamente en estos sistemas.

El economista Sendhil Mullainathan, al escribir sobre los peligros de la IA, incluyó el fenómeno de las colas largas (que expliqué en el capítulo 6) dentro de su noción de «riesgo de cola»:

> Debemos tener miedo. No de las máquinas inteligentes. Sino de las máquinas que toman decisiones para las que no tienen la inteligencia necesaria. Tengo mucho más miedo de la estupidez de las máquinas que de su inteligencia. La estupidez de las máquinas crea un riesgo de cola. Las máquinas pueden tomar muchísimas decisiones acertadas y, un día, cometer un fallo estrepitoso en una situación que no aparecía en sus datos de entrenamiento. Esa es la diferencia entre la inteligencia específica y la inteligencia general.[374]

O, como dijo el investigador de IA Pedro Domingos en una frase memorable: «A la gente le preocupa que los ordenadores se vuelvan demasiado inteligentes y se apoderen del mundo, pero el verdadero problema es que son demasiado estúpidos y ya se han apoderado del mundo».[375]

A mí me preocupa la falta de fiabilidad de la IA. También me preocupa cómo se use. Además de los aspectos éticos que abordé en el capítulo 7, una cosa concreta que me asusta es el uso de sistemas de IA para generar medios de comunicación falsos: textos, sonidos, imágenes y vídeos que muestran con un realismo aterrador hechos que en realidad nunca han sucedido.

En resumen, ¿debe aterrorizarnos la IA? Sí y no. Para las máquinas superinteligentes y conscientes falta mucho tiempo. No va a haber «todo tipo de trucos» que igualen los aspectos de nuestra humanidad que más valoramos. Por lo menos yo no lo creo. Pero sí hay muchas cosas por las que preocuparse en relación con las posibilidades de que los algoritmos y los datos se usen de forma peligrosa y poco ética. Da miedo, pero, por otro lado, es alentador ver la gran atención que este tema ha recibido en los últimos tiempos en el mundo de la IA y fuera de él. Cada vez hay más sentimiento de cooperación y propósito común entre los investigadores, las empresas y los políticos sobre la urgencia de abordar estos problemas.

Pregunta: ¿Qué problemas apasionantes de la IA siguen sin resolverse?

Casi todos.

Cuando empecé a trabajar en IA, lo que me apasionaba era, en parte, que casi todos los interrogantes estaban por resolver, a la espera de nuevas ideas. Creo que sigue siendo así.

Si nos remontamos a los inicios del sector, la propuesta de 1955 de John McCarthy y otros (descrita en el capítulo 1) enumeraba algunos de los principales temas de investigación de la IA: procesamiento del lenguaje natural, redes neuronales, aprendizaje automático, conceptos abstractos y razonamiento, y creatividad. En 2015, el director de investigación de Microsoft, Eric Horvitz, dijo en tono de broma que «incluso se podría decir que la propuesta [de 1955], debidamente reformateada, podría volverse a presentar a la National Science Foundation […] hoy mismo y probablemente conseguiría financiación de algunos gestores de programas entusiasmados».[376]

Esto no es en absoluto una crítica a las investigaciones de hace años sobre IA. La inteligencia artificial es un campo tan difícil, por lo menos, como cualquiera de los otros grandes retos científicos de la humanidad. Rodney Brooks, del MIT, lo expresó mejor que nadie: «Cuando comenzó la IA, la fuente de inspiración eran sin duda el rendimiento y la inteligencia de los seres humanos. Creo que ese objetivo es lo que atrajo a la mayoría de los investigadores a este campo durante los primeros sesenta años. El hecho de que no hayamos conseguido nada remotamente cercano a lo que se ambicionaba no significa que los investigadores no hayan trabajado mucho ni que no hayan sido excelentes. Significa que es un objetivo muy difícil». [377]

Los aspectos más apasionantes de la IA no se centran solo en las posibles aplicaciones. Quienes fundaron este campo lo hicieron porque querían resolver preguntas científicas sobre la naturaleza de la inteligencia, no solo para desarrollar nuevas tecnologías. De hecho, la idea de que la inteligencia es un fenómeno natural, que puede estudiarse como muchos otros fenómenos mediante la construcción de modelos informáticos

simplificados, fue lo que atrajo a mucha gente (yo, por ejemplo) a este campo.

La IA seguirá teniendo cada vez más repercusiones para todos nosotros. Espero que este libro les haya ayudado, como seres humanos pensantes, a tener cierta idea del estado actual de esta pujante disciplina, incluidos los numerosos problemas sin resolver, los posibles riesgos y beneficios de sus tecnologías y las preguntas científicas y filosóficas que plantea a la hora de comprender nuestra propia inteligencia humana. Y si algún ordenador está leyendo esto, que me diga quién es el sujeto de «plantea» en la frase anterior y será bienvenido a unirse a la discusión.

---

[355] «Automated Vehicles for Safety», página web de la Administración Nacional de Seguridad del Tráfico en Carretera, www.nhtsa.gov/technology-innovation/automated-vehicles-safety#issue-road-self-driving.

[356] «Vehicle Cybersecurity: DOT and Industry Have Efforts Under Way, but DOT Needs to Define Its Role in Responding to a Real-World Attack», General Accounting Office, marzo de 2016, consultado el 18 de diciembre de 2018, www.gao.gov/assets/680/676064.pdf.

[357] Citado en J. Crosbie, «Ford's Self-Driving Cars Will Live Inside Urban 'Geofences'», *Inverse*, 13 de marzo de 2017, www.inverse.com/article/28876-ford-self-driving-carsgeofences-ride-sharing.

[358] Citado en J. Kahn, «To Get Ready for Robot Driving, Some Want to Reprogram Pedestrians», *Bloomberg*, 16 de agosto de 2018, www.bloomberg.com/news/articles/ 2018-08-16/to-get-ready-for-robot-driving-some-want-to-reprogram-pedestrians.

[359] «Artificial Intelligence, Automation, and the Economy», Executive Office of the President, diciembre de 2016, https://www.whitehouse.gov/sites/whitehouse.gov/files/images/EMBARGOED%20AI%20Economy%20Report%20.pdf.

[360] Esto se remonta a lo que Alan Turing denominó «la objeción de *lady* Lovelace», en honor a *lady* Ada Lovelace, matemática y escritora británica que trabajó con Charles Babbage en el desarrollo de la Máquina Analítica, una propuesta del siglo XIX para un ordenador programable (que nunca llegó a completarse). Turing cita los escritos de *lady* Lovelace: «La Máquina Analítica no tiene pretensiones de *originar* nada. Puede hacer *cualquier cosa que sepamos ordenarle* que haga». A. M. Turing, «Computing Machinery and Intelligence», *Mind* 59, n.º 236 (1950), pp. 433-460.

[361] Página web de Karl Sims, consultada el 18 de diciembre de 2018, https://www.karlsims.com/.

[362] D. Cope, *Virtual Music: Computer Synthesis of Musical Style*, Cambridge, Mass.: MIT Press, 2004.

[363] Citado en G. Johnson, «Undiscovered Bach? No, a Computer Wrote It», *The New York Times*, 11 de noviembre de 1997.

[364] M. A. Boden, «Computer Models of Creativity», *AI Magazine* 30, n.º 3 (2009), pp. 23-34.

[365] J. Gottschall, «The Rise of Storytelling Machines», en *What to Think About Machines That Think*, ed. de J. Brockman, Nueva York: Harper Perennial, 2015, pp. 179-180.

[366] De «Creating Human-Level AI: How and When?», conferencia por vídeo, Future of Life Institute, 9 de febrero de 2017, www.youtube.com/watch?v=V0aXMTpZTfc.

[367] A. Karpathy, «The State of Computer Vision and AI: We Are Really, Really Far Away», blog de Andrej Karpathy, 22 de octubre de 2012, karpathy.github.io/2012/10/22/state-of-computer-vision.

[368] C. L. Evans, *Broad Band: The Untold Story of the Women Who Made the Internet*, Nueva York: Portfolio-Penguin, 2018, p. 24.

[369] M. Campbell-Kelly *et al.*, *Computer: A History of the Information Machine*, 3.ª ed., Nueva York: Routledge, 2018, p. 80.

[370] Citado en K. Anderson, «Enthusiasts and Skeptics Debate Artificial Intelligence», *Vanity Fair*, 26 de noviembre de 2014.

[371] Véase O. Etzioni, «No, the Experts Don't Think Superintelligent AI Is a Threat to Humanity», *Technology Review*, 20 de septiembre de 2016, www.technologyreview.com/s/602410/no-the-experts-dont-think-superintelligent-ai-is-a-threat-to-humanity; y V. C. Müller y N. Bostrom, «Future Progress in Artificial Intelligence: A Survey of Expert Opinion», en *Fundamental Issues of Artificial Intelligence*, Basilea, Suiza: Springer, 2016, pp. 555-572.

[372] N. Bostrom, «How Long Before Superintelligence?», *International Journal of Future Studies* 2 (1998).

[373] D. R. Hofstadter, *Gödel, Escher, Bach: an Eternal Golden Braid*, Nueva York: Basic Books, 1979, pp. 677-678.

[374] De «The Myth of AI: A Conversation with Jaron Lanier», *Edge*, 14 de noviembre de 2014, www.edge.org/conversation/jaron_lanier-the-myth-of-ai.

[375] P. Domingos, *The Master Algorithm*, Nueva York: Basic Books, 2015, pp. 285-286.

[376] De «Panel: Progresos en IA: mitos, realidades y aspiraciones», vídeo de Microsoft Research, consultado el 18 de diciembre de 2018, www.youtube.com/watch?v=1wPFEj1ZHRQ&feature=youtu.be.

[377] R. Brooks, «The Origins of 'Artificial Intelligence'», blog de Rodney Brooks, 27 de abril de 2018, rodneybrooks.com/forai-the-origins-of-artificial-intelligence.

Agradecimientos

Este libro debe su existencia a Douglas Hofstadter. Las obras de Doug fueron las que me atrajeron inicialmente a la IA, y sus ideas y su dirección guiaron mis estudios de doctorado. Más recientemente, Doug me invitó a la reunión en Google de la que surgió la idea de este libro, y aún más recientemente, leyó cada capítulo del manuscrito y llenó las páginas de comentarios inteligentes que mejoraron enormemente la versión final. Estoy muy agradecida a Doug por sus ideas, sus libros y artículos, su apoyo a mi trabajo y, sobre todo, su amistad.

También quiero expresar mi gratitud a otros amigos y familiares que generosamente leyeron y comentaron de forma certera cada capítulo: Jim Levenick, Jim Marshall, Russ McBride, Jack Mitchell, Norma Mitchell, Kendall Springer y Chris Wood. Muchas gracias también a las siguientes personas por responder a preguntas, traducir fragmentos y ofrecer diversos tipos de ayuda: Jeff Clune, Richard Danzig, Bob French, Garrett Kenyon, Jeff Kephart, Blake LeBaron, Sheng Lundquist, Dana Moser, David Moser y Francesca Parmeggiani.

Muchas gracias a Eric Chinski, de Farrar, Straus and Giroux, por su estímulo y sus aportaciones siempre acertadas en todos los aspectos de este proyecto; a Laird Gallagher, por las numerosas y atinadas sugerencias que ayudaron a convertir un manuscrito en bruto en un texto acabado; y al resto del equipo de FSG, especialmente a Julia Ringo, Ingrid Sterner, Rebecca

Caine, Richard Oriolo, Deborah Ghim y Brian Gittis, por todo su gran trabajo. Muchas gracias también a mi agente, Esther Newberg, por ayudarme a hacer realidad este libro.

Estoy en deuda con mi marido, Kendall Springer, por su amor constante y su apoyo entusiasta, además de su paciencia y tolerancia para con mis enloquecidos hábitos de trabajo. Mis hijos, Jacob y Nicholas Springer, han sido una maravillosa inspiración a lo largo de los años con sus atinadas preguntas, su curiosidad y su sentido común. Este libro está dedicado a mis padres, Jack y Norma Mitchell, que me han dado estímulo y amor ilimitados durante toda mi vida. En un mundo lleno de máquinas, tengo la enorme suerte de estar rodeada de unos seres humanos inteligentes y cariñosos.

Créditos de las ilustraciones

Figura 1: Dibujo de una neurona adaptado de C. Ling, M. L. Hendrickson y R. E. Kalil, «Resolving the Detailed Structure of Cortical and Thalamic Neurons in the Adult Rat Brain with Refined Biotinylated Dextran Amine Labeling», *PLOS ONE* 7, n.º 11 (2012), e45886. Licencia de uso de Creative Commons Attribution 4.0 International license (creativecommons.org/licenses/by/4.0/).

Figura 2: Imagen de los caracteres escritos a mano de Josef Steppan, commons.wikimedia.org/wiki/File:MnistExamples.png. Licencia de uso de Creative Commons Attribution-ShareAlike 4.0 International license (creativecommons.org/licenses/by-sa/4.0/deed.en).

Figura 3: Autora.

Figura 4: Autora.

Figura 5: Autora.

Figura 6: media.defense.gov/2015/May/15/2001047923/-1/-1/0/150506-F-BD468-053.JPG, visitada el 4 de diciembre de 2018 (dominio público).

Figura 7: Autora.

Figura 8: Autora.

Figura 9: Autora.

Figura 10: Autora.

Figura 11: Autora.

Figura 12: Autora.

Figura 13: Autora.

Figura 14: De twitter.com/amywebb/status/841292068488118273, visitada el 7 de diciembre de 2018. Reproducida con permiso de Amy Webb.

Figura 15: www.nps.gov/yell/learn/nature/osprey.htm (dominio público); www.fs.usda.gov/Internet/FSE_MEDIA/stelprdb5371680.jpg (dominio público).

Figura 16: De twitter.com/jackyalcine/status/615329515909156865, visitada el 7 de diciembre de 2018. Reproducida con permiso de Jacky Alcine.

Figura 17: De www.flickr.com/photos/jozjozjoz/352910684, visitada el 7 de diciembre de 2018. Reproducida con permiso de Joz Wang, de jozjozjoz.com.

Figura 18: De C. Szegedy *et al.*, «Intriguing Properties of Neural Networks», en *Proceedings of the International Conference on Learning Representations* (2014). Reproducida con permiso de Christian Szegedy.

Figura 19: De A. Nguyen, J. Yosinski y J. Clune, «Deep Neural Networks Are Easily Fooled: High Confidence Predictions for Unrecognizable Images», en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 427-436. Reproducida con permiso de los autores.

Figura 20: Figura adaptada de M. Sharif *et al.*, «Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition», en *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (2016), pp. 1528-1540. Reproducida con permiso de los autores. La fotografía de Milla Jovovich procede de commons.wikimedia.org/wiki/File:Milla Jovovich.png, de Georges Biard, con licencia de uso de Creative Commons Attribution-Share Alike 3.0 Unported license (creativecommons.org/licenses/by-sa/3.0/deed.en).

Figura 21: Autora.

Figura 22: De www.cs.cmu.edu/~robosoccer/image-gallery/legged/2003/aibo-with-ball12.jpg. Reproducida con permiso de Manuela Veloso.

Figura 23: Autora.

Figura 24: Autora.

Figura 25: Autora.

Figura 26: Autora.

Figura 27: Autora.

Figura 28: Autora.

Figura 29: Autora.

Figura 30: Autora.

Figura 31: Autora.

Figura 32: Autora.

Figura 33: Autora.

Figura 34: Autora.

Figura 35: Autora.

Figura 36: Autora.

Figura 37: De T. Mikolov *et al.*, «Distributed Representations of Words and Phrases and Their Compositionality», en *Advances in Neural Information Processing Systems* (2013), pp. 3111-3119. Reproducida con permiso de Tomas Mikolov.

Figura 38: Autora.

Figura 39: Autora. La fotografía procede de la base de datos de Microsoft COCO: cocodataset.org.

Figura 40: La fotografía y los pies de foto son de la base de datos de Microsoft COCO: cocodataset.org.

Figura 41: Las fotografías y los pies de foto son de nic.droppages.com. Reproducidas con permiso de Oriol Vinyals.

Figura 42: *Fila superior*: fotografías y pies de foto de O. Vinyals *et al.*, «Show and Tell: A Neural Image Caption Generator», en *Proceedings of*

the IEEE Conference on Computer Vision and Pattern Recognition (2015), pp. 3156-3164. Reproducidas con permiso de Oriol Vinyals. *Fila inferior, izquierda*: Road-Tech Safety Services. Reproducida con permiso de Ben Jeffrey. *Fila inferior, derecha*: Nikoretro, https://www.flickr.com/photos/bellatrix6/4727507323/in/album-72057594083648059. Licencia de Creative Commons Attribution-ShareAlike 2.0 Generic license https://creativecommons.org/licenses/by-sa/2.0/. Los pies de foto de la fila inferior son de captionbot.ai.

Figura 43: Fotografías y pies de foto de H. Chen *et al.*, «Attacking Visual Language Grounding with Adversarial Examples: A Case Study on Neural Image Captioning», en *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, vol. 1, *Long Papers* (2018), pp. 2587-2597. Reproducidas con permiso de Hongge Chen y la Asociación de Lingüística Computacional.

Figura 44: Dorothy Alexander / Alamy Stock Photo.

Figura 45: De www.foundalis.com/res/bps/bpidx.htm. Las imágenes originales son de M. Bongard, *Pattern Recognition*, Nueva York: Spartan Books, 1970.

Figura 46: De www.foundalis.com/res/bps/bpidx.htm. Las imágenes originales son de M. Bongard, *Pattern Recognition*, Nueva York: Spartan Books, 1970.

Figura 47: Autora.

Figura 48: Fotografías hechas por la autora.

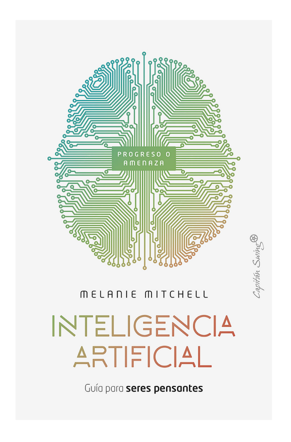
Figura 49: www.nps.gov/dena/planyourvisit/pets.htm (dominio público); pxhere.com/en/photo/1394259 (dominio público); Peter Titmuss /Alamy Stock Photo; Thang Nguyen, www.flickr.com/photos/70209763@N00/399996115, licencia de Creative Commons Attribution-ShareAlike 2.0 Generic license (creativecommons.org/licenses/by-sa/2.0/).

Figura 50: P. Souza, *Obama: An Intimate Portrait*, Nueva York: Little, Brown, 2018, p. 102 (dominio público).

Índice

Créditos

Inteligencia artificial



Melanie Mitchell separa la ciencia real de la ciencia ficción en este amplio examen del estado actual de la IA y de cómo está rehaciendo nuestro mundo.

Ninguna empresa científica reciente ha resultado tan seductora, aterradora y llena de extravagantes promesas y frustrantes reveses como la inteligencia artificial. La galardonada autora Melanie Mitchell, una destacada científica informática, revela ahora la turbulenta historia de la IA y la reciente oleada de aparentes éxitos, grandes esperanzas y temores emergentes que la rodean.

En 'Inteligencia Artificial', Mitchell aborda las cuestiones más urgentes de la IA en la actualidad: ¿Hasta qué punto son realmente inteligentes los mejores programas de IA? ¿Cómo funcionan? ¿Qué pueden hacer realmente y cuándo fallan? ¿Hasta qué punto esperamos que se asemejen a los humanos y cuándo debemos preocuparnos de que nos superen? Por el camino, presenta los modelos dominantes de la IA y el aprendizaje automático modernos, describiendo los programas de IA más avanzados, sus inventores humanos y las líneas de pensamiento históricas que sustentan los logros recientes. Se reúne con otros expertos como Douglas Hofstadter, científico cognitivo y autor del clásico moderno Gödel, Escher, Bach, ganador del Premio Pulitzer, quien explica por qué está "aterrorizado" ante el futuro de la IA. Explora la profunda desconexión entre el bombo publicitario y los logros reales de la IA, proporcionando una idea clara de lo que el campo ha logrado y cuánto le queda por recorrer.

Entrelazando historias sobre la ciencia de la IA y la gente que hay detrás, ‘Inteligencia Artificial’ rebosa de relatos claros, cautivadores y accesibles de los trabajos modernos más interesantes y provocativos en este campo, aderezados con el humor y las observaciones personales de Mitchell.

Este libro franco y animado es una guía indispensable para entender la IA actual, su búsqueda de una inteligencia "de nivel humano" y su impacto en el futuro de todos nosotros.

Un bienvenido correctivo a los temores y esperanzas exagerados sobre la IA, y el manual prefecto para empezar a entender cómo funcionan realmente los sistemas." -Alison Gopnik, profesora de Psicología en la Universidad de Berkeley "Sin rehuir los detalles técnicos, este estudio ofrece un curso accesible sobre redes neuronales, visión por ordenador y procesamiento del lenguaje natural, y plantea si la búsqueda de una inteligencia general abstraída es preocupante… La visión de Mitchell es tranquilizadora". -*The New Yorker*

"Mitchell se desmarca de la exageración a la que a menudo es propenso el campo de la inteligencia artificial y expone lo que hace bien, en qué falla y cómo podría hacerlo mejor". -*George Musser*.

Melania Mitchell. Catedrática Davis de Complejidad en el Instituto Santa Fe. Sus principales trabajos se han desarrollado en los ámbitos del razonamiento analógico, los sistemas complejos, los algoritmos genéticos y los autómatas celulares, y sus publicaciones en esos campos se citan con frecuencia. Estudió Física, Astronomía y Matemáticas en la Universidad Brown. Su interés por la inteligencia artificial se despertó en la universidad cuando leyó Gödel, Escher, Bach, de Douglas Hofstadter. Tras licenciarse trabajó como profesora de matemáticas en un instituto de Nueva York. Decidió que "tenía que dedicarse" a la inteligencia artificial y buscó a Douglas Hofstadter para pedirle en varias ocasiones ser una de sus estudiantes de posgrado. Finalmente consiguió un puesto de becaria para trabajar en el desarrollo de *Copycat*. Obtuvo su doctorado en 1990. Participa regularmente como experta invitada en el Learning Salon, un encuentro interdisciplinar en línea sobre inteligencia biológica y artificial. En 2020, Mitchell recibió el premio Herbert A. Simon.

Título original: *Artificial Intelligence: A Guide for Thinking Humans* (2020)

Edición en ebook: abril de 2024

c/ Rafael Finat 58, 2º 4 - 28044 Madrid
Tlf: (+34) 630 022 531
28044 Madrid (España)
contacto@capitanswing.com
www.capitanswing.com

ISBN: 978-84-127797-4-5

Diseño de colección: Filo Estudio - www.filoestudio.com
Corrección ortotipográfica: Victoria Parra Ortiz
Composición digital: leerendigital.com